



Ektron CMS400.NET Developer Reference Manual

Version 7.0, Rev 8

Ektron® Inc.
542 Amherst St.
Nashua, NH 03063
Tel: +1 603-594-0249
Fax: +1 603-594-0258
Email: sales@ektron.com
<http://www.ektron.com>

Copyright 2007 Ektron®, Inc. All rights reserved.

EKTRON is a registered trademark of Ektron, Inc.

Version 7.0, Rev 8, June 26, 2007

For updated versions of this manual, please visit:
<http://www.ektron.com/web-content-management-solutions.aspx?id=4020>

EKTRON, INC. SOFTWARE LICENSE AGREEMENT

**** READ THIS BEFORE LOADING SOFTWARE****

YOUR RIGHT TO USE THE PRODUCT DELIVERED IS SUBJECT TO THE TERMS AND CONDITIONS SET OUT IN THIS LICENSE AGREEMENT. USING THIS PRODUCT SIGNIFIES YOUR AGREEMENT TO THESE TERMS. IF YOU DO NOT AGREE TO THIS SOFTWARE LICENSE AGREEMENT, DO NOT DOWNLOAD.

CUSTOMER should carefully read the following terms and conditions before using the software program(s) contained herein (the "Software"). Downloading and/or using the Software or copying the Software onto CUSTOMER'S computer hard drive indicates CUSTOMER'S acceptance of these terms and conditions. If CUSTOMER does not agree with the terms of this agreement, CUSTOMER should not download.

Ektron, Inc. ("Ektron") grants, and the CUSTOMER accepts, a nontransferable and nonexplosive License to use the Software on the following terms and conditions:

1. Right to use: The Software is licensed for use only in delivered code form. Each copy of the Software is licensed for use only on a single URL. Each license is valid for the number of seats listed below (the "Basic Package"). Any use of the Software beyond the number of authorized seats contained in the Basic Package without paying additional license fees as provided herein shall cause this license to terminate. Should CUSTOMER wish to add seats beyond the seats licensed in the Basic Package, the CUSTOMER may add seats on a block basis at the then current price for additional seats (see product pages for current price). The Basic Packages are as follows:

Ektron CMS300 - Licensed for ten seats (10 named users) per URL.

Ektron CMS400.NET - Licensed for ten seats (10 named users) per URL.

Ektron eWebEditPro - Licensed for ten seats (10 named users) per URL.

Ektron eWebEditPro+XML - Licensed for ten seats (10 named users) per URL.

For purposes of this section, the term "seat" shall mean an individual user provided access to the capabilities of the Software.

The CUSTOMER may not modify, alter, reverse engineer, disassemble, or decompile the Software. This software product is licensed, not sold.

2. Duration: This License shall continue so long as CUSTOMER uses the Software in compliance with this License. Should CUSTOMER breach any of its obligations hereunder, CUSTOMER agrees to return all copies of the Software and this License upon notification and demand by Ektron.

3. Copyright: The Software (including any images, "applets," photographs, animations, video, audio, music and text incorporated into the Software) as well as any accompanying written materials (the "Documentation") is owned by Ektron or its suppliers, is protected by United States copyright laws and international treaties, and contains confidential information and trade secrets. CUSTOMER agrees to protect the confidentiality of the Software and Documentation. CUSTOMER agrees that it will not provide a copy of this Software or Documentation nor divulge any proprietary information of Ektron to any person, other than its employees, without the prior consent of Ektron; CUSTOMER shall use its best efforts to see that any user of the Software licensed hereunder complies with this license.

4. Limited Warranty: Ektron warrants solely that the medium upon which the Software is delivered will be free from defects in material and workmanship under normal, proper and intended usage for a period of three (3) months from the date of receipt. Ektron does not warrant the use of the Software will be uninterrupted or error free, nor that program errors will be corrected. This limited warranty shall not apply to any error or failure resulting from (i) machine error, (ii) Customer's failure to follow operating instructions, (iii) negligence or accident, or (iv) modifications to the Software by any person or entity other than Company. In the event of a breach of warranty, Customer's sole and exclusive remedy, is repair of all or any portion of the Software. If such remedy fails of its essential purpose, Customer's sole remedy and Ektron's maximum liability shall be a refund of the paid purchase price for the defective Products only. This limited warranty is only valid if Ektron receives written notice of breach of warranty within thirty days after the warranty period expires.

5. Limitation of Warranties and Liability: THE SOFTWARE AND DOCUMENTATION ARE SOLD "AS IS" AND WITHOUT ANY WARRANTIES AS TO THE PERFORMANCE, MERCHANTABILITY, DESIGN, OR OPERATION OF THE SOFTWARE. NO WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE IS OFFERED. EXCEPT AS DESCRIBED IN SECTION 4, ALL WARRANTIES EXPRESS AND IMPLIED ARE HEREBY DISCLAIMED.

NEITHER COMPANY NOR ITS SUPPLIERS SHALL BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS OR GOODWILL, LOSS OF DATA OR USE OF DATA, INTERRUPTION OF BUSINESS NOR FOR ANY OTHER INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND UNDER OR ARISING OUT OF, OR IN ANY RELATED TO THIS AGREEMENT, HOWEVER, CAUSED, WHETHER FOR BREACH OF WARRANTY, BREACH OR REPUDIATION OF CONTRACT, TORT, NEGLIGENCE, OR OTHERWISE, EVEN IF COMPANY OR ITS REPRESENTATIVES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSS.

6. Miscellaneous: This License Agreement, the License granted hereunder, and the Software may not be assigned or in any way transferred without the prior written consent of Ektron. This Agreement and its performance and all claims arising from the relationship between the parties contemplated herein shall be governed by, construed and enforced in accordance with the laws of the State of New Hampshire without regard to conflict of laws principles thereof. The parties agree that any action brought in connection with this Agreement shall be maintained only in a court of competent subject matter jurisdiction located in the State of New Hampshire or in any court to which appeal therefrom may be taken. The parties hereby consent to the exclusive personal jurisdiction of such courts in the State of New Hampshire for all such purposes. The United Nations Convention on Contracts for the International Sale of Goods is specifically excluded from governing this License. If any provision of this License is to be held unenforceable, such holding will not affect the validity of the other provisions hereof. Failure of a party to enforce any provision of this Agreement shall not constitute or be construed as a waiver of such provision or of the right to enforce such provision. If you fail to comply with any term of this License, YOUR LICENSE IS AUTOMATICALLY TERMINATED. This License represents the entire understanding between the parties with respect to its subject matter.

YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, THAT YOU UNDERSTAND THIS AGREEMENT, AND UNDERSTAND THAT BY CONTINUING THE INSTALLATION OF THE SOFTWARE, BY LOADING OR RUNNING THE SOFTWARE, OR BY PLACING OR COPYING THE SOFTWARE ONTO YOUR COMPUTER HARD DRIVE, YOU AGREE TO BE BOUND BY THIS AGREEMENT'S TERMS AND CONDITIONS. YOU FURTHER AGREE THAT, EXCEPT FOR WRITTEN SEPARATE AGREEMENTS BETWEEN EKTRON AND YOU, THIS AGREEMENT IS A COMPLETE AND EXCLUSIVE STATEMENT OF THE RIGHTS AND LIABILITIES OF THE PARTIES.

Copyright 1999 - 2007 Ektron, Inc. All rights reserved.

LA10031, Revision 1.7

Securing Ektron CMS400.NET

Security Checklist

The following is a checklist of things an Administrator must do to secure **Ektron CMS400.NET**.

☐ **Change Admin user password**

☐ **Change builtin user password**

IMPORTANT!

If you changed the builtin user password during the site setup, you do not need to change it again. See "Editing Builtin" on page ii for additional information.

☐ **Remove Sample Users and Sample Membership Users**

IMPORTANT!

Some users in this list might not appear in your User list. This depends on which version of the software you have installed.

CMS400.NET Users	Membership Users
<input type="checkbox"/> jedit <input type="checkbox"/> Admin2 <input type="checkbox"/> tbrown <input type="checkbox"/> jsmith <input type="checkbox"/> vs <input type="checkbox"/> EkExplorerUser <input type="checkbox"/> Explorer	<input type="checkbox"/> jmember <input type="checkbox"/> member@example.com <input type="checkbox"/> north <input type="checkbox"/> supermember <input type="checkbox"/> west

Additional Information

You *do not* need to make these changes if you are using the CMS400.NET for demonstration or evaluation purposes. These changes should be completed once the decision is made to purchase the product and go live with your site.

IMPORTANT!	Ektron recommends creating your own Administrator user and deleting the Admin user.
IMPORTANT!	Ektron recommends deleting any users from the CMS400.NET that are not needed on the system.
IMPORTANT!	By default, when the Ektron Explorer client is installed, the Explorer or EkExplorerUser username and password are also installed. Client-side users should use their Ektron CMS400.NET username and password when using the Ektron Explorer. To learn how to change this information, see the Administrator Manual section "Ektron Explorer" > "Configuring Ektron Explorer."



The Everyone Group

By default, the root folder in Workarea provides the Everyone Group with all permissions except Overwrite Library. Ektron Inc. recommends reviewing the permission needs of the Everyone Group when you add folder. For information on Permissions, see the Administrator Manual section "Managing User and Permissions" > "Setting Permissions."

Editing Builtin

IMPORTANT!	The "builtin" user does not appear in the Users list. This user appears on the application setup screen.
-------------------	--

To change the builtin account information, follow these steps.

1. In the Workarea, click **Settings > Configuration > Setup**.
2. Click the Edit button (.
3. Locate the **Built In User** field.
4. Edit the built in user information by changing the username and password.
5. Click the Update button (.

NOTE If you cannot sign in to **Ektron CMS400.NET** because the builtin user password was changed and you don't know the new password, use the **BuiltinAccountReset.exe** utility. This resets your **Ektron CMS400.NET** user \ password to **Builtin \ Builtin**. This utility is located in `C:\Program Files\Ektron\CMS400versionnumber\Utilities`.

ServerControlWS.asmx

Best Practice

For improved security, Ektron recommends renaming or removing the Web services file when you move it to your production server. After installation, this file is named **ServerControlWS.asmx** and resides in the *webroot/siteroot/Workarea* folder.

ServerControlWS.asmx is the Web service that lets the server controls talk to **Ektron CMS400.NET**. The path is coded in the `web.config` file. This is how it appears:

```
<!-- Web Service URL for Server Controls design time --> <add key="WSPath" value="http://localhost/CMS400Min/Workarea/ServerControlWS.asmx" />
```

You will need to edit this line if you change the location or name of the **ServerControlWS.asmx** file.

What's New in the 7.0 Release

- Ektron has introduced its own markup language, the Ektron Markup Language (EkML), to help developers define the look and feel of a server control when displayed a Web page. See Also: ["Ektron Markup Language" on page 754](#)
- Added four new proxies to the list of Web proxies available when using Plug-in Extensions. See Also: ["Connecting to the Web Service API From a Plug-in" on page 836](#)
 - GetPermissionsProxy() creates a connection to the Web Service API Permissions.asmx.
 - GetTaskProxy() creates a connection to the Web Service API Task.asmx.
 - GetTaskCategoryProxy() creates a connection to the Web Service API TaskCategory.asmx
 - GetTaskCategoryTypeProxy() creates a connection to the Web Service API TaskCategoryType.asmx
- A new Web Search server control is available. The previous search server control is deprecated. See Also: ["Web Search Server Control" on page 306](#)
- A new Directory server control lets you determine most aspects of the Taxonomy feature. See ["Directory Server Control" on page 141](#)
- Added two new Discussion Board server controls, ActiveTopics and PostHistory. The ActiveTopics server control displays the most active or most recent topics. See Also: ["ActiveTopics Server Control" on page 155](#). The PostHistory server control displays a list of posts for a given user. See Also: ["PostHistory Server Control" on page 157](#)
- A new Theme property is available in the Forum server control. This property allows you to specify a custom location for your themes. See Also: ["Using a Custom Theme" on page 150](#)
- Added the DynamicParameter property to the Forum and Blog server controls. This property allows a board's ID or blog's ID to be passed dynamically to its control. See Also: ["Forum Server Control Properties" on page 148](#) and ["Blog Server Control Properties" on page 55](#)
- A new HideUserProfile property is available for the Forum server control. When set to True, this property hides all user profiles. See Also: ["Forum Server Control Properties" on page 148](#)
- An ExcludeList property has been added to the Forum server control. A comma separated list of custom user properties can be excluded from the display of the Forum's profile page. ["Forum Server Control Properties" on page 148](#)

- A Map server control lets you display a map with links to content. For example, if your Web site manages a state university system, a map can show the location of each campus. See Also: ["Map Server Control" on page 190](#)
- A Flexible Menu server control provides new capabilities not available in previous menus. See Also: ["Flexible Menu Server Control" on page 232](#)
- The Blog server control has a new property that allows you to define the number of posts visible. See the MaxResults property in ["Blog Server Control Properties" on page 55](#)
- A BlogArchive server control has been added to the Blog server controls. See Also: ["BlogArchive Server Control" on page 70](#)
- A new ContentReview server control is available. The ContentRating control is deprecated. See Also: ["ContentReview Server Control" on page 127](#)

Summary Table of Contents

Securing Ektron CMS400.NET	i
What's New in the 7.0 Release	v
About This Manual	1
List of Developer Resources	2
Introduction to Ektron CMS400.NET Server Controls	7
AssetControl Server Control.....	51
Blog Server Controls	54
BreadCrumb Server Control	72
FolderBreadcrumb Server Control	82
BusinessRules Server Control	86
Calendar Server Control	88
Collection Server Control	95
Content Analytics Server Control.....	106
ContentBlock Server Control.....	108
ContentList Server Control	117
ContentReview Server Control	127
ContentRating Server Control	134
DesignTimeDiagnostic Server Control	139
Directory Server Control	141
Discussion Board Server Controls.....	147
FormBlock Server Control	159
HTMLEditor Server Control.....	163
ImageControl Server Control	171
LanguageAPI Server Control	173
LanguageSelect Server Control.....	176
ListSummary Server Control	178
Login Server Control	186
Map Server Control	190
Membership Server Control	214
Introduction to Menu Server Controls	223
Smart Menu Server Control.....	226

Flexible Menu Server Control	232
Menu Server Control.....	246
DhtmlMenu Server Control.....	254
MetaData Server Control	261
MetaDataList Server Control	266
Personalization Server Controls.....	276
Poll Server Control.....	298
RssAggregator Server Control	302
Web Search Server Control.....	306
Search Server Control	318
IndexSearch Server Control.....	351
SiteMap Server Control	368
 Dreamweaver Extension Support	 372
 Standard ASP.NET Methods	 596
 Custom ASP Functions	 604
Login Function	607
Content Block Function.....	609
Add Content.....	617
Search Functions	621
Search Display Function	631
List Summary Function	635
XML List Summary Function.....	642
Archive List Summary Function	647
Single Summary Function.....	648
RSS Summary Function	652
Collection Function.....	657
RSS Collection Function	661
XML Collection Function	663
Random Content	669
Random Summary	670
Event Calendar Function.....	671
Calendar Function.....	674
Form Function	678
MultiPurpose Function	680
Metadata Function.....	681
Show Site Language Function.....	687
DHTML Menus	688

XML Menu	692
Custom ColdFusion Functions	694
Custom PHP Functions	714
Custom API.....	735
Custom JSP Functions	736
Ektron Markup Language	754
Ektron's Developer SDK	805
Plug-in Extension	813
Web Services	856
Syndicating Web Content as RSS	876
Syndicating via .NET.....	877
Learning About Visual Studio	879
Customizing Validation.....	880
Programmatic Search API	898
Cascading Style Sheets	900
Additional Information	943

Detailed Table of Contents

Securing Ektron CMS400.NET	i
Security Checklist.....	i
Additional Information	ii
The Everyone Group	ii
Editing BuiltIn.....	ii
ServerControlWS.aspx	iii
 What's New in the 7.0 Release	v
 About This Manual	1
 List of Developer Resources	2
Ektron's DevCenter.....	2
User Discussion Boards	2
Ektron Blogs	2
Ektron FileShare.....	3
Tutorials	3
External Resources	3
Knowledge Base Articles.....	3
Ektron's Developer Training Course.....	3
Developer Site Provided with Ektron CMS400.NET Install	3
Ektron CMS400.NET API Documentation	4
API Help Inside Visual Studio 2005	4
API Help in PDF Format.....	5
From the Desktop	5
From the Workarea	5
From www.ektron.com	5
Ektron Support and Product Pages	5
Supporting Documents	6
 Introduction to Ektron CMS400.NET Server Controls	7
Opening the Sample Project.....	7
Setting Up a Template	9
Making the Server Controls Available	10
Removing the Server Controls	12
Updating the Server Controls	12
Inserting Server Controls Using Drag and Drop.....	12
Adding a Server Control to a Template.....	12

ServerControlWS.aspx.....	14
Using CMS Explorer to Browse Your Ektron CMS400.NET Site...	15
Build the Project	16
Inserting Server Controls Programmatically.....	16
Steps to Insert a Server Control Programmatically.....	17
Step 1: Declare the Namespace	17
Step 2: Create an Instance of the New Control	18
Step 3: Set the Properties	18
Step 4: Call the Fill Method	19
Step 5: Use .text Property to Display the Control.....	19
Server Control Properties in Code Behind	20
Displaying Properties for a Control	20
Mutual Server Control Properties in Code Behind	21
Accessing Additional Properties.....	23
Accessing Items in an Array.....	30
Referencing the Page Property	30
Customizing Server Controls.....	31
Recognizing the Server Control Within the HTML	31
Recognizing the Server Control Within the Code Behind	31
Customizing the Server Control in the Code Behind.....	32
Troubleshooting Error Creating Control Message.....	32
Using Drag & Drop and Programmatically Together	33
Data Binding with Server Controls.....	35
Steps to Data Binding Using Drag and Drop Server Controls	36
Caching with Server Controls	42
Caching Individual Content.....	43
Caching While not Logged In	43
Caching While Logged In	43
Setting up Caching of Individual Content	43
Server Controls that Support Caching of Individual Content.....	44
Page Level Caching	45
Using Ajax Enabled Server Controls and Custom Code.....	46
List of Server Controls	46
AssetControl Server Control.....	51
AssetControl Server Control Properties.....	51
Blog Server Controls	54
Using the Blog Server Control vs. Using the Individual Blog	
Server Controls.....	54
Blog Server Control.....	54
Blog Server Control Properties.....	55
BlogEntries Server Control.....	59
BlogEntries Server Control Properties.....	59
BlogPost Server Control	61
BlogPost Server Control Properties.....	61
BlogCalendar Server Control	63

BlogCalendar Server Control Properties	63
BlogRoll Server Control	64
BlogRoll Server Control Properties.....	65
BlogCategories Server Control	65
BlogCategories Server Control Properties	66
BlogRecentPosts Server Control	67
BlogRecentPosts Server Control Properties	67
BlogRSS Server Control	68
BlogRSS Server Control Properties	68
BlogArchive Server Control.....	70
BlogArchive Server Control Properties.....	70
BreadCrumb Server Control	72
BreadCrumb Properties Apply to Web Forms, Not Web Pages	
74	
Types of BreadCrumb Server Control Properties.....	74
Properties that Determine a Breadcrumb Trail's Appearance	74
Properties that Determine How Pages Associated with the Form	
Appear on a Breadcrumb Trail.....	75
BreadCrumb Server Control Properties	75
BreadCrumb Metadata Type	79
Using the BreadCrumb Server Control.....	79
Making a Content Block's Title Appear in the Breadcrumb Trail	
80	
Steps to Use a Content Block's Title as a Breadcrumb	80
FolderBreadcrumb Server Control	82
BreadCrumb Server Control Properties	82
Using the BreadCrumb Server Control.....	84
BusinessRules Server Control	86
BusinessRules Server Control Properties	86
Calendar Server Control.....	88
Calendar Server Control Properties.....	89
Calendar Style Sheet	93
Collection Server Control	95
Collection Server Control Properties.....	95
Example of ecmNavigation Display	101
ecmNavigation XSL code.....	101
Example of ecmTeaser Display	102
ecmTeaser XSL code	102
Retrieving the XML Structure of a Collection.....	103
Using the Collection Server Control Programmatically Exam-	
ple.....	104
Content Analytics Server Control.....	106
Content Analytics Server Control Properties.....	106
Ektron Properties.....	106
ContentBlock Server Control.....	108

Static Content Block	108
ContentBlock Server Control Properties.....	108
Dynamic Content Block	110
XML Content Block	112
The OverrideXslt and DisplayXslt Properties	113
Retrieving the XML Structure of an XML Content Block.....	115
Using the ContentBlock Server Control Programmatically Ex- ample	115
ContentList Server Control	117
ContentList Server Control Properties	117
Using the DefaultContentID and MetaTag Properties.....	122
Using the ContentID Property to Display a Content List.....	126
ContentReview Server Control	127
ContentReview Server Control Properties	127
Using the DisplayXSLT Property	129
Displaying Content Reviews On the Site	131
Displaying a User's Reviews On the Site	132
ContentRating Server Control	134
Content Rating Server Control Properties	134
Ektron Properties.....	134
Ektron Label Properties	136
Initial Graph Display	137
Complete Graph Display	137
DesignTimeDiagnostic Server Control	139
DesignTimeDiagnostic Server Control Properties	140
Directory Server Control	141
Improving Taxonomy Performance	141
Directory Server Control Properties	141
Discussion Board Server Controls	147
Forum Server Control	147
Forum Server Control Properties.....	148
Updating the Page Command	150
Using a Custom Theme.....	150
Allowing HTML Editing in the Javascript Editor	151
Adding and Removing Toolbar Buttons From the Javascript Editor	151
ActiveTopics Server Control	155
ActiveTopics Server Control Properties	155
PostHistory Server Control	157
PostHistory Server Control Properties.....	157
FormBlock Server Control	159
FormBlock Server Control Properties	160
Automatic versus Manual Generation of Form Tags	161
HTMLEditor Server Control	163
Introduction	163

Working with Ektron CMS400.NET Content	164
Working with Non Ektron CMS400.NET Content	165
Uploading Files with an Action Page.....	165
HTMLEditor Server Control Properties.....	165
ShowCMSToolbar	169
ToolbarLevel	169
ImageControl Server Control	171
ImageControl Server Control Properties	171
LanguageAPI Server Control	173
LanguageSelect Server Control vs. LanguageAPI Server Control	173
LanguageAPI Server Control Properties	173
Code Behind Only Properties and Methods.....	174
Using the LanguageAPI Server Control Programmatically ..	175
LanguageSelect Server Control.....	176
LanguageSelect Server Control vs. LanguageAPI Server Control	176
LanguageSelect Server Control Properties.....	176
ListSummary Server Control	178
ListSummary Server Control Properties	178
OrderKey Property.....	184
Retrieving the XML Structure of a List Summary	185
Login Server Control	186
Placing a Login Button.....	186
Login Server Control Properties	187
Map Server Control.....	190
Introduction.....	190
Using Maps on the Sample Site.....	191
Obtaining a License to Use Google Maps	192
Changing the Map's Zoom Level	194
Using the Search Tab to Recenter the Map.....	194
Using the Find What Tab to Find Locations with a Search Term ...	194
Using the Directions Tab.....	195
The Display on Map Button	195
.....	196
Displaying Icons on the Category Popup Screen.....	197
Text Box.....	197
Bubble Text.....	199
Controlling the Visitor Experience with Maps.....	200
Displaying/Suppressing Map Elements	200
Determining a Map's Initial Boundaries	201
Determining which Content is Found by a Map.....	201
Preparing CMS Content to Appear on a Map	201
Restricting Content for a Particular Map	204
Map Server Control Properties.....	204

Membership Server Control	214
Membership Server Control Properties	215
DisplayMode.....	219
Updating a Membership User's Information.....	220
ShowTerms.....	221
Introduction to Menu Server Controls	223
Prerequisites	223
Contrasting Menu Server Controls	223
Smart Menu Server Control.....	226
Smart Menu Server Control Properties.....	226
Section 508 Compliance Features of Smart Menus.....	230
Flexible Menu Server Control	232
Sample Menus.....	233
Description of the Sample Menus.....	234
Copying and Editing Sample Menus	237
Working with the Flex Menu Xslt File.....	238
Setting up a Master/Slave Menu Relationship	239
Flex Menu Server Control Properties.....	240
Menu Server Control.....	246
Menu Server Control Properties.....	246
Using DisplayXslt Samples.....	248
SampleMenu	248
SampleMenu XSL code	250
TreeMenu	250
TreeMenu XSL code	252
Retrieving the XML Structure of a Menu.....	252
DhtmlMenu Server Control.....	254
DhtmlMenu Server Control Properties.....	255
Using a Style Sheet	260
Formatting the DhtmlMenu Server Control	260
MetaData Server Control	261
MetaData Server Control vs. MetaDataList Server Control..	261
MetaData Server Control Properties	261
Using the Simple Dublin Core Metadata Standard	263
Using the MetaData Server Control.....	264
MetaDataList Server Control	266
MetaData Server Control vs. MetaDataList Server Control..	266
MetaDataList Server Control Properties	266
Retrieving the XML Structure of a MetadataList	275
Personalization Server Controls.....	276
Microsoft WebParts	277
EktronCatalogPart Server Control	279
EktronCatalogPart Server Control Properties.....	279
EktronCommunityCatalogPart Server Control.....	282

EktronCommunityCatalogPart Server Control Properties.....	282
EktronWebPartZone Server Control	283
EktronWebPartZone Server Control Properties	284
PersonalizationManager Server Control.....	284
PersonalizationManager Server Control Properties	285
Building a Personalization Web Page.....	289
Poll Server Control.....	298
Poll Server Control Properties	298
Automatic versus Manual Generation of Form Tags	300
RssAggregator Server Control	302
RSSAggregator Server Control Properties	302
Using the RssAggregator Server Control.....	304
Web Search Server Control.....	306
Property Usage Table	306
Property Descriptions	307
ResultTagId.....	313
Displaying WebSearch Results on a Separate Page	314
Code Example: Displaying WebSearch Results on a Separate Page.....	316
Additional Resources.....	316
Using a Style Sheet	316
Search Server Control	318
Introduction.....	318
Search Server Control vs. IndexSearch Server Control.....	319
Search Server Control Properties	319
Property Usage Table	320
Property Descriptions	321
Ajax Search	332
ResultTagId	332
Text Box Examples	333
Button Image Source Examples	334
Search Button Text Examples	335
Display Examples	335
ShowDate Examples	336
ShowExtendedSearch	336
Using the Advanced Search Properties.....	339
Basic and Standard Searches	340
Advanced Search	341
Using Standard and Advanced Searches Together	342
Fieldset Properties	345
Retrieving the XML Structure of a Search	347
Displaying Search Results on a Separate Page.....	347
Using a Style Sheet	349
Formatting the Search Server Control.....	350
IndexSearch Server Control.....	351

IndexSearch Server Control vs. Search Server Control.....	351
IndexSearch Server Control Properties.....	351
SearchParmXML Property.....	358
Using the IndexSearch Server Control Programmatically Ex- ample	361
Programmatically Predefined General Search Results Re- placed By Specific Search Results	364
Turn Off Caching In Netscape Navigator and Mozilla Firefox....	367
SiteMap Server Control	368
Sitemap Server Control Properties	368
Using the Sitemap Server Control	370
 Dreamweaver Extension Support	372
Requirements	372
Preparing Dreamweaver for Ektron CMS400.NET Functions	372
Installing the Extensions.....	373
Setup.....	377
Editing and Saving Ektron CMS400.NET Content in Dream- weaver	381
Accessing Manage Content Commands... ..	381
Edit Content.....	382
Save Content.....	385
Saving New Content	385
Saving Existing Ektron CMS400.NET Content	388
Using the Ektron CMS400.NET Extensions.....	389
Custom Function Symbols.....	390
Ektron CMS400.NET Command Menus.....	395
Inserting Custom Functions	402
CMS Function.....	402
ActiveTopics.....	402
Analytics	405
AssetControl	408
Blog	411
Blog Archive	415
Blog Calendar.....	418
Blog Categories	421
Blog Entries	424
Blog Post	428
Blog Recent Posts	431
Blog Roll	433
Blog RSS	436
BreadCrumb	439
Calendar	445

Collection	449
Content Block	456
Content List	460
Content Rating.....	466
Content XSLT Tag... ..	470
DHTML Menu	471
Directory-Taxonomy.....	478
Discussion Forum.....	485
Display By Meta Value.....	487
Dynamic Content Block	495
Folder Bread Crumb	497
Form	499
ImageControl	503
Language API.....	506
Language Select Box	509
List Summary.....	511
Login.....	520
Map Control	525
Membership Control	532
Metadata for ASP.NET Pages	537
Metadata Function for ASP, CF, JSP or PHP Pages	540
Multipurpose Content Block.....	543
Poll	545
PostHistory	548
Random Content	551
Random Summary.....	553
RSS Aggregator	554
Old Search.....	557
Search Display	568
Single Summary	570
Site Map	573
Smart Menu	576
Standard Menu	580
Web Search.....	583
XML Index Search	589
 Standard ASP.NET Methods	 596
Replacement of Standard Methods	596
Referencing Methods in Visual Basic	597
Adding a New Web Form.....	597
Requirements for Standard Methods	598
Standard Methods.....	599
Addcontent.....	600
Add Content Example	602
 Custom ASP Functions	 604

Login Function	607
Login Tag Example	607
Placing Your Login Button.....	608
Content Block Function.....	609
Content Block Examples.....	611
Static Content Block	612
Dynamic Content Block	612
XML Content Block.....	614
The Function's Arguments	614
Dynamic and Static Combination.....	616
Add Content.....	617
Add Content Example	618
Search Functions	621
Search Tag Example	623
Search Tag Attributes.....	624
Starting Folder	624
Recursive.....	624
Target Page	624
Text Box.....	625
Maximum Characters	625
Button Image Source.....	626
Button Text.....	627
Font Face	627
Font Size	629
Horizontal	630
Spare 1	630
Search Display Function	631
Search Display Example	631
Search Display with Default Content	632
Search Display Attributes	632
Max Number of Content Blocks Returned	632
Style Info.....	633
Show Date	634
List Summary Function	635
List Summary Example	636
List Summary Attributes	637
Starting Folder	637
Recursive.....	638
Show Summary	638
Style Info.....	639
Order By	639
Ascending/Descending	640
Show Info.....	640
Max Number.....	641
XML List Summary Function.....	642
Archive List Summary Function	647

Single Summary Function	648
Single Summary Example	648
Single Summary Attributes	649
Content ID	649
Show Summary	649
Style Info	650
Show Info	650
Spare 1	651
RSS Summary Function	652
RSS Summary Example	652
RSS Summary Attributes	655
Starting Folder	655
Recursive	655
Order By	655
Ascending/Descending	656
Max Number	656
Collection Function	657
Collection Examples	657
The ecmNavigation Display Function	657
The ecmTeaser Display Function	658
Collection Attributes	660
ID	660
DisplayFunction	660
RSS Collection Function	661
XML Collection Function	663
Random Content	669
Random Summary	670
Event Calendar Function	671
Event Calendar Style Sheet	671
Event Calendar Sample	671
Calendar Function	674
Calendar Style Sheet	676
Event Calendar Sample	676
Form Function	678
MultiPurpose Function	680
Metadata Function	681
Creating an Item List	681
ecmMetadata Example	682
More ecmMetadata Examples	683
Using a Dynamic Metadata Tag	684
ecmMetadata Attributes	684
Item List	684
Spare 1	686
Show Site Language Function	687

DHTML Menus	688
Additional Documentation in Ektron Knowledge Base	688
Displaying a Menu on a Web Page	688
GenerateMenu and DropMenu	688
Displaying Menus in a Multi-language System	689
Customizing the Menu's Appearance	689
ek_Menu.js	689
FontFamily	691
menuLiteBgColor	691
XML Menu	692
 Custom ColdFusion Functions	 694
Login Function	696
Login Tag Example	696
Content Block Function	696
Static Content Block	697
Dynamic Content Block	697
XML Content Block	697
Add Content	697
Add Content Example	698
Random Content Function	698
Random Summary Function	699
Search Function	699
Search Tag Example	699
Search Display Function	700
Search Display Example	700
List Summary Function	700
List Summary Example	700
XML List Summary Function	701
XML List Summary Example	701
RSS List Summary Function	701
List Summary Example	701
Archive List Summary Function	702
Single Summary Function	702
Single Summary Example	702
Collection Function	702
Example Using the ecmNavigation Display Function	703
Example Using the ecmTeaser Display Function	704
Collection Variables	705
DisplayFunction	705
XML Collection Function	706
RSS Collection Function	706
Event Calendar Function	706
Event Calendar Sample	709
Form Function	709

MultiPurpose Function	709
Metadata Function	710
DHTML Menus	711
Additional Documentation in Ektron Knowledge Base	712
Displaying a Menu on a Web Page	712
GenerateMenu and DropMenu	712
Customizing the Menu's Appearance	712
ek_Menu.js.....	713
XML Menu Function	713
Language Select Function	713

Custom PHP Functions714

Login Function	717
Login Function Example	717
Content Block Function	717
Static Content Block	718
Dynamic Content Block	718
XML Content Block.....	718
Add Content	718
Add Content Example.....	719
Search Function	719
Search Function Example	720
Search Display Function	720
Search Display Function Example.....	720
List Summary Function	720
List Summary Example.....	721
Archive List Summary Function	721
Single Summary Function	721
Single Summary Example	721
XML List Summary Function	722
RSS List Summary Function	722
Collection Function	722
Collection Examples	723
Example Using the ecmNavigation Display Function.....	723
Example Using the ecmTeaser Display Function.....	724
Collection Variables	725
DisplayFunction	725
XML Collection Function	725
RSS Collection Function	726
Random Content Function	726
Random Summary Function	726
Event Calendar Function	727
Event Calendar Sample.....	729
Form Function	729
MultiPurpose Function	730

Metadata Function	730
DHTML Menus	732
Additional Documentation in Ektron Knowledge Base	733
Displaying a Menu on a Web Page	733
GenerateMenu and DropMenu	733
Customizing the Menu's Appearance	733
ek_Menu.js.....	734
XML Menu Function.....	734
Language Select Function	734
Custom API.....	735

Custom JSP Functions	736
Calendar Function	738
Calendar Sample.....	739
Event Calendar Function	739
Event Calendar Sample.....	739
Collection Function	740
Collection Sample.....	740
XML Collection Function.....	740
Random Content Function.....	741
Random Summary Function.....	741
RSS Collection Function	741
Add Content	742
Add Content Example.....	742
Content Block Function	742
Static Content Block	743
Dynamic Content Block	743
XML Content Block.....	743
Form Function.....	743
Language Select Function	745
Login Function	745
Login Function Example	746
DHTML Menus	746
Additional Documentation in Ektron Knowledge Base	746
Displaying a Menu on a Web Page	746
GenerateMenu and DropMenu	746
Customizing the Menu's Appearance	747
ek_Menu.js.....	747
XML Menu Function.....	747
Metadata Function	748
Search Function.....	750
Search Function Example	751
Search Display Function.....	751
Search Display Function Example.....	751
List Summary Function.....	752

List Summary Example.....	752
XML List Summary Function	752
RSS List Summary Function	752
Single Summary Function	753
Single Summary Example	753

Ektron Markup Language754

Introduction.....	754
EkML Templates.....	754
EkML Example	755
EkML Tags	758
EkML Variables	758
EkML Variables.....	758
[\$Comment]	763
[\$ContentId]	763
[\$DateCreated].....	764
[\$DateModified]	765
[\$EditorFirstName]	765
[\$EditorLastName]	766
[\$FolderId].....	766
[\$Html].....	766
[\$HyperLink].....	767
[\$Image]	768
[\$ImageIcon]	768
[\$ImageThumbnail]	769
[\$Index]	769
[\$ItemCount]	769
[\$Language]	769
[\$LinkTarget]	770
[\$QuickLink]	770
[\$SERVER_NAME]	771
[\$ShowAllcategory]	771
[\$ShowBubble].....	772
[\$ShowBubble(width)]	773
[\$ShowContent('htmltagid')]	773
[\$Status]	774
[\$Teaser]	774
[\$Title]	775
[\$UrlParam('paramname')].....	776
[\$UrlEncode('str')]	777
EkML Template Examples	778
collection.ekml	778
collection.ekml Variables	778
[\$CollectionDescription]	780
[\$CollectionTitle].....	781
contentlist.ekml.....	781

contentlist.ekml Variables.....	781
listsummary.ekml	783
ListSummary.ekml Variables	784
[\$FolderDescription]	786
[\$FolderName]	787
map.ekml	787
__Map	789
__SearchTxtResultPane	790
__RouteInfoPane	791
metadatalist.ekml.....	791
metadatlist.ekml Variables	791
taxonomy.ekml.....	794
Taxonomy <ekoutput> modes	794
taxonomy.ekml Variables	795
[\$TemplateQuickLink].....	798
websearch.ekml.....	798
websearch.ekml Variables	799
[\$SearchSummary]	800
[\$ContentByteSize]	801
[\$PagingCurrentEndIndex].....	801
[\$ItemCount]	802
[\$SearchText]	803
[\$PagingCurrentStartIndex].....	803
[\$SearchDuration]	804
Ektron's Developer SDK	805
Installing the Developer SDK.....	806
Using the Ektron Site Setup Wizard.....	806
Plug-in Extension	813
Introduction.....	813
How a Plug-in Works with CMS400.NET	813
The Plug-in Extension Architecture and the Plug-in Wizard	813
Managing your Plug-ins.....	813
Creating a Plug-in Overview	814
Installing the Developer SDK.....	814
Creating a New Plug-in Extension in Visual Studio 2005.....	815
Plug-in Samples.....	819
PublishSendAlert Code Sample	820
PublishContentChange Sample	821
InstantMessageOnPublish Sample	822
Manually Creating Plug-in Framework.....	823
Configuring a Plug-in Extension	823
Opening the Extensibility Configuration Editor	824
Viewing Plug-in Extensions in the Extensibility Configuration Editor...	824

The Extensibility Configuration Editor's Toolbar	826
Setting Plug-in Extension and Event Properties	826
Property List	829
Applying Plug-ins and Events to Specific Sites	830
Determining What Happens After an Error Occurs	832
Setting the Information Properties in the Plug-in Code	833
Enabling and Disabling Plug-ins and Events	834
Setting the Plug-in Order of Execution by Event	835
Connecting to the Web Service API From a Plug-in	836
Debugging a Plug-in	837
Debugging a C# Plug-in Extension	838
Debugging a VB Plug-in Extension	847
 Web Services	 856
Introduction to Web Services	856
CMS400.NET's Implementation of Web Services	856
Benefits of Using Web Services	856
WS Components and CMS400.NET Architecture	857
Software and Hardware Requirements	857
Installation of Files	857
Setup Instructions	857
Providing Web Service Information in Your CMS	859
CMS400.NET Methods	859
The CMS400.NET Content Sample Page	862
The Search Display Method Parameters	865
Sample Output from One Method	867
GetContentBlock	867
The News.aspx Sample (non-XML)	868
Identifying the Code Behind File	868
Setting Up an Instance of Web Services	868
Getting Content Blocks	868
The News.aspx Sample (XML)	870
Setting Up Instances of Web Services	870
Getting Content Blocks	870
Creating a Page that Provides a Web Service	874
 Syndicating Web Content as RSS	 876
What is RSS Syndication?	876
The RSS Specification	876
RSS Syndication in Ektron CMS400.NET	876
Syndicating via .NET	877
Architecture of RSS Syndication	877
aspx pages	877
Retrieving a Collection or List Summary as an RSS Feed	877
Creating Your Own aspx Collection Page	877

Example.....	877
Remarks.....	877
Example.....	878
Creating Your Own aspx List Summary Page	878
Example.....	878
Remarks.....	878
Example.....	878
Learning About Visual Studio	879
Grid Layout vs. Flow Layout.....	879
Customizing Validation.....	880
Customizing Validation Options.....	880
Validation Elements in the Configuration Data	880
Validation Attributes.....	881
Validation Sub-elements	881
Choice Sub-element	881
Choice Attributes.....	882
Choice Sub-Elements	883
Defining a Schema Fragment	883
Defining a Calculation	884
Defining Validation	885
Defining an Error Message	886
Defining Custom Validation	886
Saving Invalid Documents	890
Custom Script that Handles Saving Invalid Files	890
The onerror Event	891
Calculated Fields	891
Icons on the Select Field or Group Screen.....	891
Learn More about XPath	893
XPath Operators.....	893
XPath Functions	894
XPath References.....	897
Programmatic Search API	898
Guidelines for Creating a Search	899
Cascading Style Sheets.....	900
reportchart.css.....	900
blogs.css.....	904
Layout Styles.....	904
Header Styles.....	905
Entry Styles	906
Calendar Styles.....	906
Blogroll Styles	909

Blog Categories Styles	910
Blog Recent Posts Styles	911
calendarStyles.css	912
Single Day View (Day)	913
Month View (month)	916
Inline View (inline)	928
Month-Up with Event (monthupwithevent)	930
Event Type Control (used in month display)	937
designformentry.css	940
pop_style.css	940
CMS400Demo.css	941
Developer Styles.css	941
Default.css	941
List Summary OrderedList.css	941
List Summary unorderedlist.css	941
Collection OrderedList.css	941
Collection unorderedlist.css	941
CSSMenu.css	941
Search unorderedList.css	941
Discussion Board CSS Files	941

Additional Information	943
Supporting Documents	943
Support	943
Sales	943
Information	943

About This Manual

This manual can be used as a reference for the developer, or the person who is setting up your Ektron CMS400.NET Web site.

To utilize the full potential of an Ektron CMS400.NET driven Web site, learning and using the server controls supplied by Ektron is essential. These server controls help make your site more robust and easier to maintain. For more information, see ["Introduction to Ektron CMS400.NET Server Controls" on page 7](#).

This document explains how to manually use custom tags for each platform supported by Ektron CMS400.NET:

- ["Standard ASP.NET Methods" on page 596](#)
- ["Custom ASP Functions" on page 604](#)
- ["Custom ColdFusion Functions" on page 694](#)
- ["Custom PHP Functions" on page 714](#)

Besides explaining how to insert the functions to your Web templates manually, this manual also explains how to create Ektron CMS400.NET templates in a WYSIWYG environment by using the Ektron supplied server controls for Visual Studio 2005 or by using the Dreamweaver extension supplied with the installation. See ["Dreamweaver Extension Support" on page 372](#) for additional information.

Ektron CMS400.NET has also implemented support for Web Services as well as syndication using RSS. Information about configuration and use of these features is also found in the following sections:

- ["Web Services" on page 856](#)
- ["Syndicating Web Content as RSS" on page 876](#)

List of Developer Resources

In addition to the information in this manual, the following resources provide information on developing your site. Below is a quick reference list of resources.

- "Ektron's DevCenter" on page 2
 - <http://dev.ektron.com>
- "Ektron's Developer Training Course" on page 3
 - <http://www.ektron.com/cms400-web-cms.aspx?id=2653>
- "Developer Site Provided with Ektron CMS400.NET Install" on page 3
 - <http://localhost/CMS400developer/default.aspx>
- "Ektron CMS400.NET API Documentation" on page 4
 - <http://www.ektron.com/web-content-management-solutions.aspx?id=4020>
- "Ektron Support and Product Pages" on page 5
 - http://www.ektron.com/support/cms400_support.aspx
 - <http://www.ektron.com/web-content-management-solutions.aspx>
- "Supporting Documents" on page 6
 - <http://www.ektron.com/web-content-management-solutions.aspx?id=4020>

Ektron's DevCenter

- <http://dev.ektron.com>

The Ektron DevCenter includes a wide array of resources to assist our developers and partners. Whether you're looking for a solution to a specific problem or just seeing what you might be able to do with CMS400.NET, you should be able to find what you need here.

User Discussion Boards

- <http://dev.ektron.com/forum.aspx>

Use the Forum to exchange ideas and information about Ektron, Inc. products.

Ektron Blogs

- <http://dev.ektron.com/blogs.aspx>

With Blogs, learn what developers and engineers at Ektron, Inc. are thinking.

Ektron FileShare

- <http://dev.ektron.com/fileshare.aspx>

The Ektron FileShare is a system designed to allow developers to share server controls, code snippets, classes and other tools that may be useful to other developers.

Tutorials

- <http://dev.ektron.com/tutorials.aspx>

This area provides video tutorials to help you learn how to use CMS400.

External Resources

- <http://dev.ektron.com/template.aspx?id=2756>

The sites listed on this page contain tutorials, developer news, or other materials that might be useful to you while developing your CMS400.NET site. These sites are not run by Ektron, Inc.

Knowledge Base Articles

- http://dev.ektron.com/kb_search.aspx

These articles provide information about all Ektron, Inc. products.

Ektron's Developer Training Course

- <http://www.ektron.com/cms400-web-cms.aspx?id=2653>

Why should you become a Certified Ektron Developer?

- Learn best practices for deploying an Ektron CMS400.NET driven Web site
- Learn how to extend CMS400.NET to do more of what you want
- See real-world examples of how an Ektron CMS is deployed
- Be an expert on the most commonly deployed CMS
- Add “Certified Ektron Developer” to your resume

Developer Site Provided with Ektron CMS400.NET Install

After installing **Ektron CMS400.NET**, you need to install the Developer Sample site, by downloading and running CMS400Developer.exe. Next you can access the sample via this url: <http://localhost/CMS400developer/default.aspx>

NOTE Replace localhost with location of your site.

WARNING!

To enable the execution of some of the API samples on the developer's Web site, set the "ek_executeDevSamples" value to **True** in the Web.Config file. If this key is set to false (default value is false), you will get a message saying "Permission denied."

Ektron has compiled a list of several different examples of developing your Web site with Ektron CMS400.NET. These examples show how easy it is to create robust Web pages by using "out-of-the-box" functionality including server controls.

Ektron CMS400.NET API Documentation

The API documentation provides a way to learn about the methods and functions that are available when customizing your Ektron CMS400.NET site. The following APIs are documented:

- Server Control API
- Business API

Access the API documentation from within Visual Studio 2005 or in PDF format.

API Help Inside Visual Studio 2005

NOTE

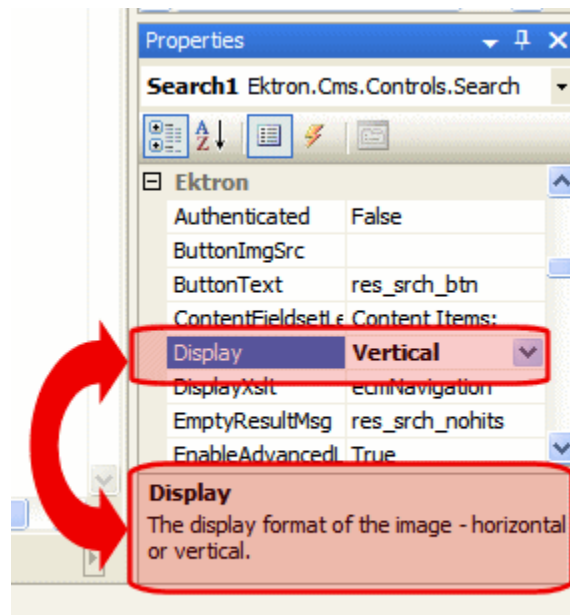
Ektron API Documentation is only available inside VS2005 if it was on the server when Ektron CMS400.NET was installed. Otherwise, use the PDF version of the help.

For API Help Inside Visual Studio 2005, follow these steps:

1. In Visual Studio 2005 click **Help > Contents**
2. When the help window appears, expand Ektron CMS400.NET **API Documentation** in the frame on the left.

If you do not see this entry, use the filter and choose **(unfiltered)** or **Ektron Documentation**.

NOTE In addition to the API help, Server Control property descriptions are shown when you highlight a property in design view in VS2005.



API Help in PDF Format

From the Desktop

Click **Start > Programs > Ektron > CMS400 > Documentation > API Reference Manual**

From the Workarea

Click **Settings > Help > Developer's API Reference Manual**

From www.ektron.com

<http://www.ektron.com/web-content-management-solutions.aspx?id=4020>

Ektron Support and Product Pages

- http://www.ektron.com/support/cms400_support.aspx
- <http://www.ektron.com/web-content-management-solutions.aspx>

Ektron's Support and Product pages are a great launch pad to learn about Ektron CMS400.NET.

Supporting Documents

You can find more detailed information about Ektron CMS400.NET in the following manuals:

- Ektron CMS400.NET Administrator Manual
- Ektron CMS400.NET User Manual
- Ektron CMS400.NET Quick Start Manual
- Ektron CMS400.NET Setup Manual

Introduction to Ektron CMS400.NET Server Controls

Ektron CMS400.NET's server controls let you insert many standard methods and properties within the Visual Studio 2005 environment. This means that you can see the effect of your changes in real time -- you don't have to modify a page then compile a sample project to see the results.

You can insert server controls using drag and drop or programmatically. You can also use databinding to retrieve and display data from **Ektron CMS400.NET**. This chapter explains the use of the Ektron CMS400 Server Controls through the following topics.

- "Opening the Sample Project" on page 7
- "Setting Up a Template" on page 9
- "Making the Server Controls Available" on page 10
- "Inserting Server Controls Using Drag and Drop" on page 12
- "Inserting Server Controls Programmatically" on page 16
- "Using Drag & Drop and Programmatically Together" on page 33
- "Data Binding with Server Controls" on page 35
- "Caching with Server Controls" on page 42
- "Using Ajax Enabled Server Controls and Custom Code" on page 46
- "List of Server Controls" on page 46

Additional information and examples for Ektron Server Controls are available online after you install the Developer sample site. Here is the path:

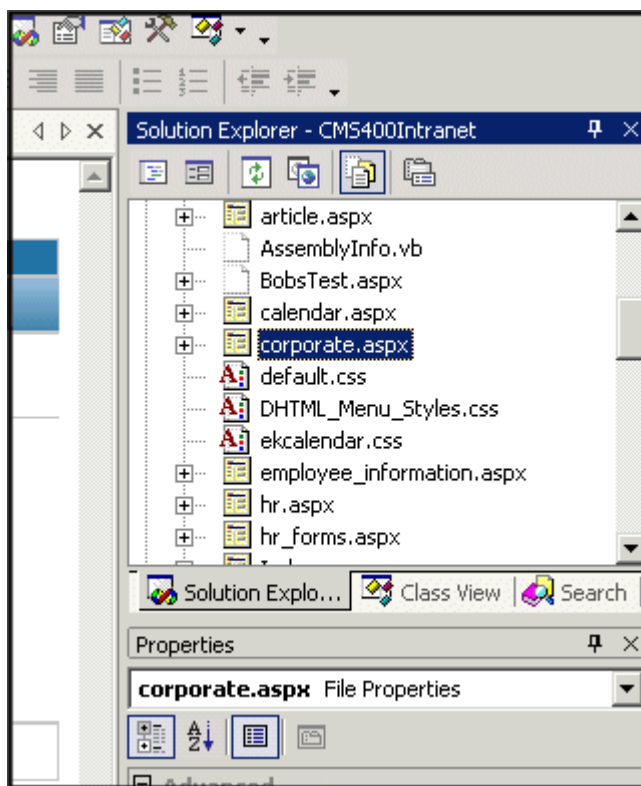
- <http://localhost/cms400Developer/Default.aspx>

Replace `localhost` with the `webroot` where you installed the Developer site.

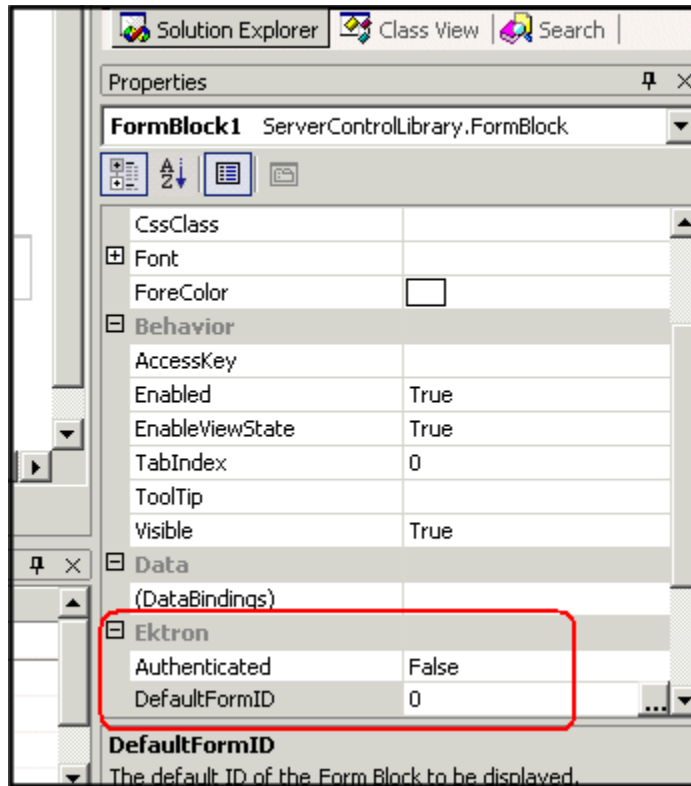
Opening the Sample Project

To help get you started, these directions explain how to open the sample site delivered with **Ektron CMS400.NET**.

1. Browse to and double click Ektron CMS400.NET's solution file, `localhost/siteroot/cms400Min.sln`.
2. The sample site project opens.
3. To work on a template page, click it from the Solution Explorer.



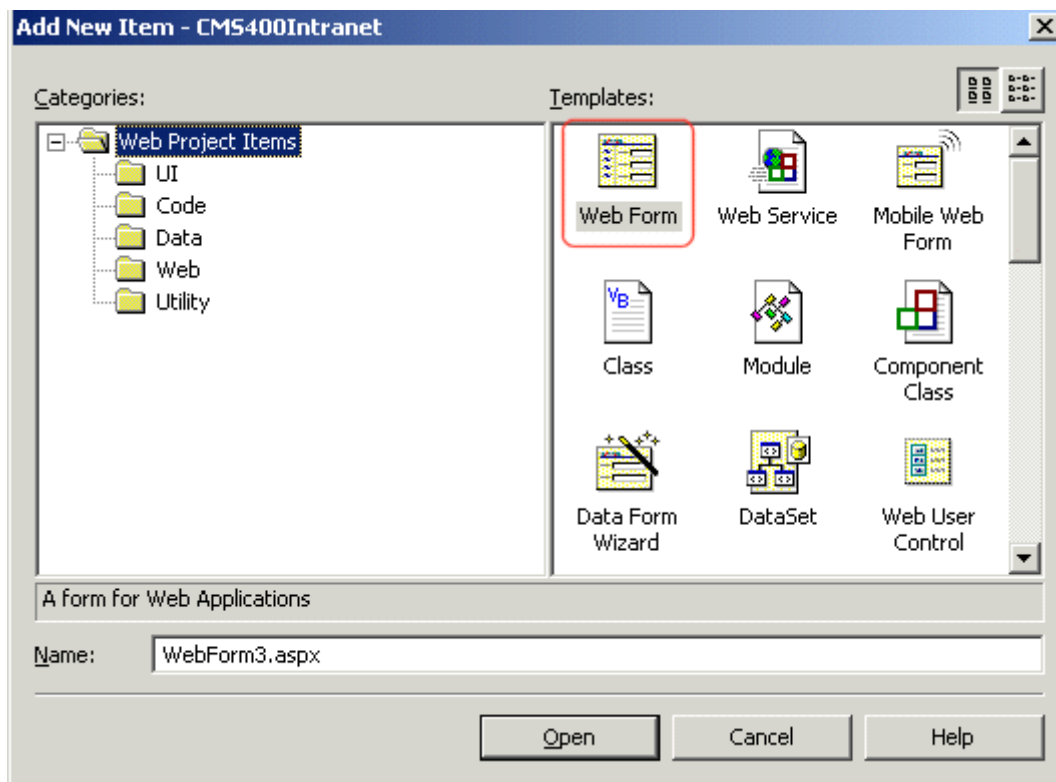
When you select a page, its properties appear in the Properties area (below Solution Explorer), and the page appears in the center of the screen. A control's properties include several standard .NET properties along with Ektron-specific ones. The Ektron properties are labeled as illustrated below.



Setting Up a Template

To create a new template (.aspx) page, follow these steps.

1. Click **Project > Add Web Form**.
2. On the Add New Item screen, click Web Form and assign a name.



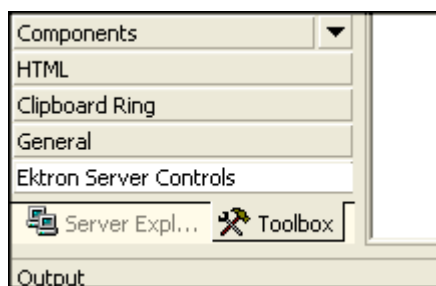
3. Add controls to determine the page content.

Making the Server Controls Available

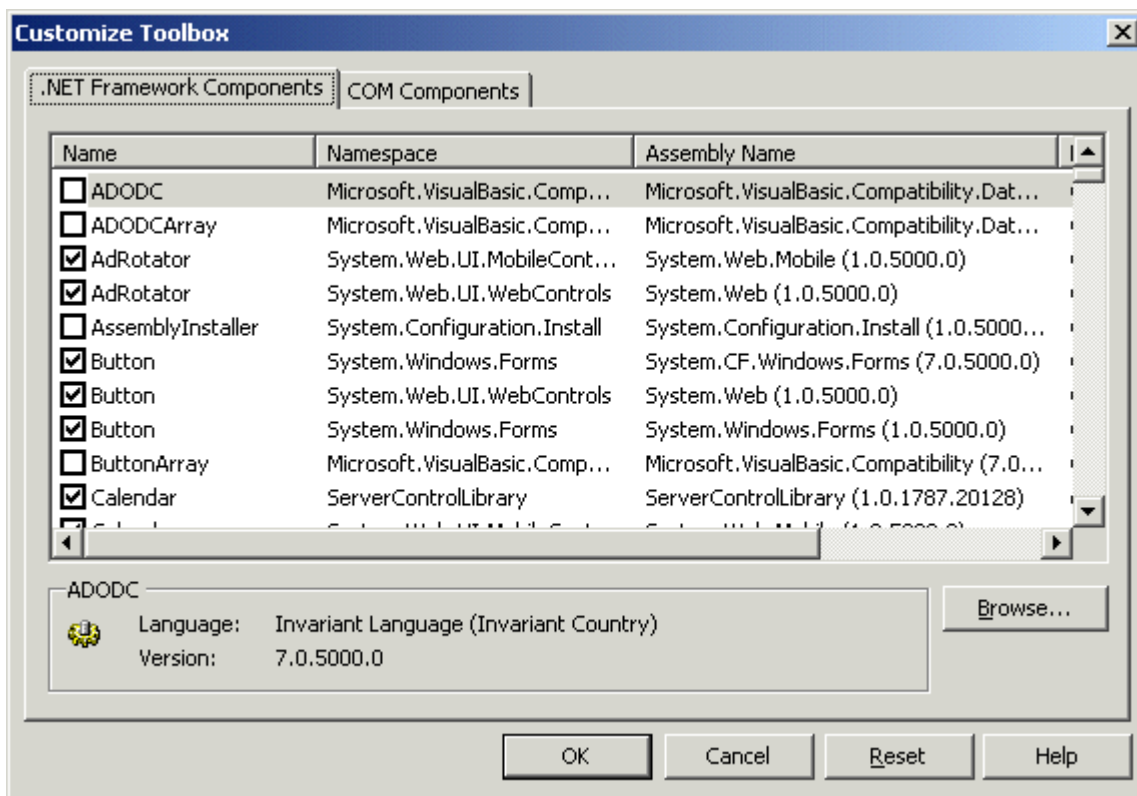
NOTE You must copy the dlls to a local drive before installing them. You cannot copy them from a network drive.

1. Display the Visual Studio 2005 toolbox (**View > Toolbox**).
2. Right click the mouse within the Toolbox.
3. Click **Add Tab**.

4. Type **Ektron Server Controls** then press enter.



5. Click the Ektron Server Controls Tab.
6. Right click the mouse in the empty area.
7. Click **Add/Remove Items**. If using Visual Studio 2005, click **Choose Items**.
8. The Customize Toolbox dialog appears.



9. Select the **.NET Framework Components** tab.

10. Browse to the directory that stores Ektron CMS400.NET's dll files, `localhost/cms400Min/bin`, and add the `Ektron.Cms.Controls.dll` file. This file provides access to **Ektron CMS400.NET**'s server controls.
Alternatively, you could use the following location, `C:\Program Files\Ektron\CMS400v7x\bin`. The file is identical in both places.
Using the bin folder in your site provides better speed when loading Web pages. However, if you use the bin folder located in Program Files, you do not have to worry about deleting the .dll file if you change or delete your site.
11. Press **OK**.

For easier viewing once the server controls are installed, you can right click on them and select Sort Items Alphabetically. Note that you can only see the server controls when an aspx template is selected.

Removing the Server Controls

1. Display the Visual Studio 2005 toolbox (**View > Toolbox**).
2. Right click the mouse within the Toolbox.
3. Click **Add/Remove Items**. If using Visual Studio 2005, click **Choose Items**.
4. Click **Namespace** or **Assembly Name** to sort the Server Controls by manufacturer.
5. Uncheck all boxes that are `Ektron.Cms.Controls`.
6. Click **OK**
7. Right click the Ektron Server Control Tab.
8. Click **Delete Tab** to remove it.

Updating the Server Controls

To update the server controls you must first remove the existing server controls in Visual Basic, then add the new server controls.

For information on removing server controls, see ["Removing the Server Controls" on page 12](#)

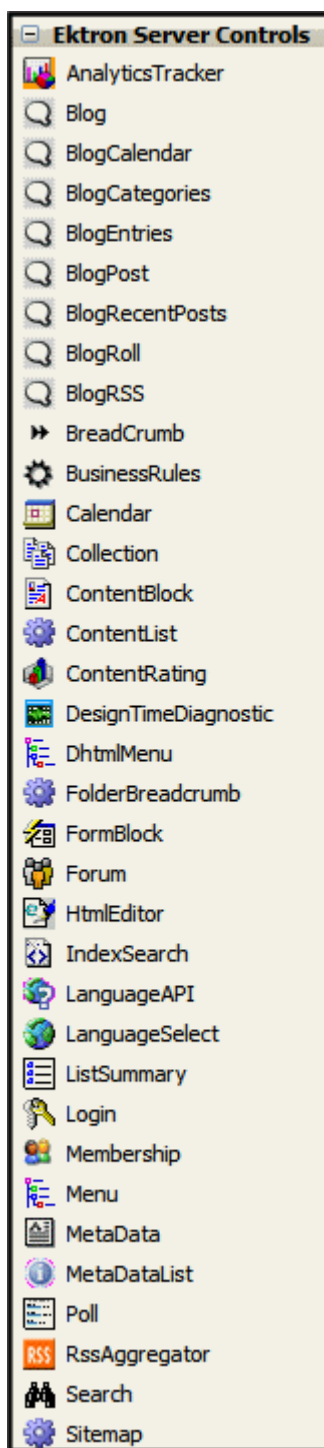
For information on adding server controls, see ["Making the Server Controls Available" on page 10](#)

Inserting Server Controls Using Drag and Drop

Because Visual Studio 2005 is a visual environment, you can watch the page layout change as you add or move a control and adjust its properties. And, whenever you need to identify an **Ektron CMS400.NET** object (such as content block or collection), you can use the CMS Explorer to visually browse to and select it.

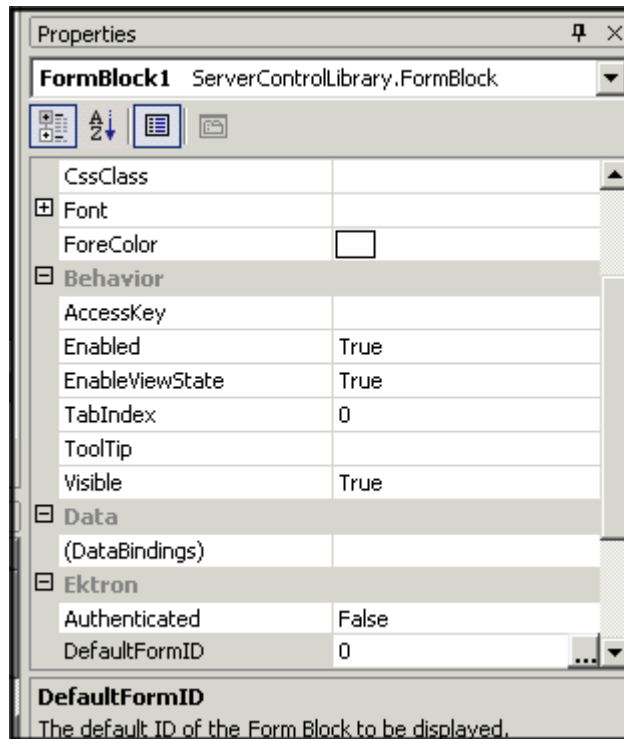
Adding a Server Control to a Template

1. Display the Visual Studio 2005 toolbox (**View > Toolbox**).
2. Click the **Ektron Server Control** tab. The **Ektron CMS400.NET**'s server controls appear.



3. Drag a server control and drop it into the desired location on the page.

- As desired, modify the control's properties using the Properties area of Visual Studio 2005.



For a description of server control properties, see ["List of Server Controls" on page 46](#).

ServerControlWS.asmx

Best Practice

For improved security, Ektron recommends renaming or removing the Web services file when you move it to your production server. After installation, this file is named `ServerControlWS.asmx` and resides in the `/siteroot/Workarea/` folder in your Web root.

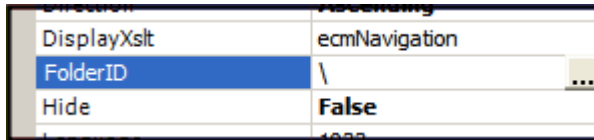
`ServerControlWS.asmx` is the Web service that lets the server controls talk to **Ektron CMS400.NET**. The path is coded in the web.config file. This is how it appears:

```
<!-- Web Service URL for Server Controls design time --> <add key="WSPath" value="http://localhost/siteroot/Workarea/ServerControlWS.asmx" />
```

You will need to edit this line if you change the location or name of the `ServerControlWS.asmx` file.

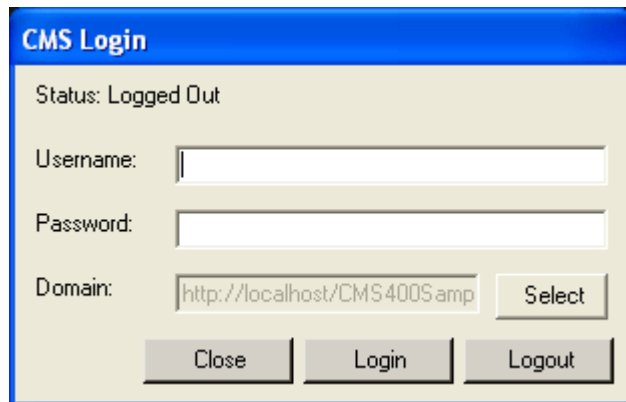
Using CMS Explorer to Browse Your Ektron CMS400.NET Site

Ektron CMS400.NET provides a window (called the CMS Explorer) that lets you browse through your Web site to identify CMS objects such as folders, calendars, blogs and content blocks. For example, if you insert a List Summary Server Control, its **FolderID** property identifies the folder whose contents are displayed. If you are not sure of the folder path, click the gray button.

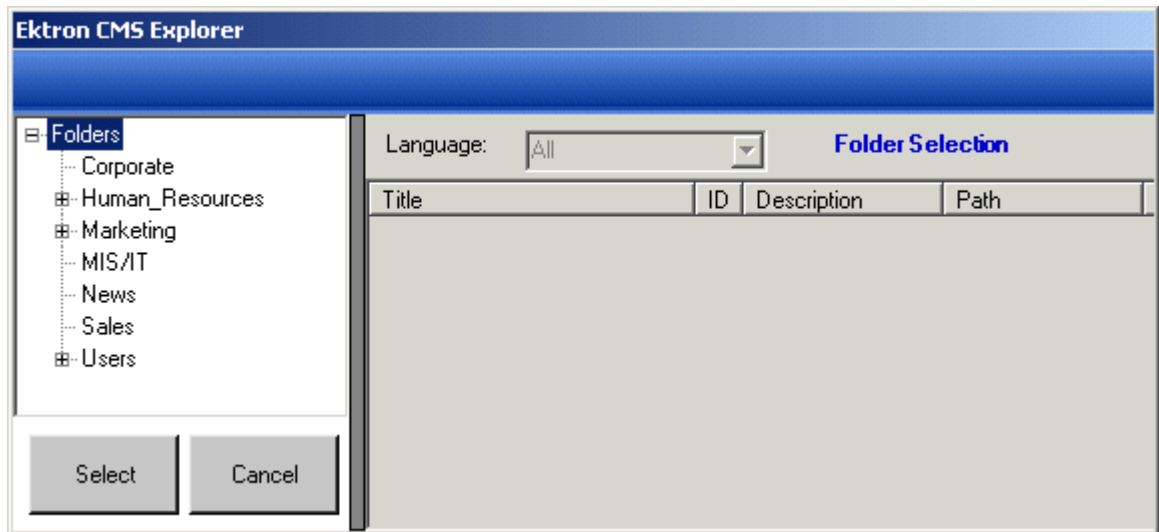


When you do, a login screen appears (illustrated below).

NOTE A user cannot be logged into the CMS Explorer and the CMS400.NET Workarea at the same time. If you log into the CMS Explorer while logged into the Workarea, you will receive the following error when you try to use the Workarea: Error: The current user does not have permission to carry out this request [CBRR31]. If you log into the Workarea while logged into the CMS Explorer, you will need to re-login to the CMS Explorer when you return to using it.



Enter your **Ektron CMS400.NET** username and password. When you do, the CMS Explorer window appears, providing a visual way to browse your **Ektron CMS400.NET** Web site.



Navigate through the folders, select an object, then click the **Select** button. The selected object is pasted to the server control property.

NOTE Although you see the object in the selected language in Visual Studio 2005, the language is not stored. For example, if you select a German content block whose ID=2, **Ektron CMS400.NET** only stores content block ID=2. When a visitor to your site browses to that page, the content block does not appear in the selected language. Instead, it is determined by a cookie or the user's language selection.

Build the Project

After you open the solution file in Visual Studio 2005 and add the required files, build the project. This may take a while the first time you do so. After the project is built, a browser opens and it is displayed as a Web page.

You can also view a Web page while working on it by right clicking on the Web form and clicking **View in Browser**.

Inserting Server Controls Programmatically

Besides using the drag-and-drop method to insert a server control (explained in ["Inserting Server Controls Using Drag and Drop" on page 12](#)), you can insert one programmatically. You might prefer to insert server controls programmatically for several reasons, such as

- you want the control to be loaded into memory only under certain conditions. In this case, insert the logic that only displays the control if the condition exists.

- you want to display only certain properties of an object, such as the title of last edited date of a content block.

Note that you can drag and drop controls on your page then customize them using code behind. For more information, see: ["Using Drag & Drop and Programmatically Together" on page 33](#).

The following sections explain how to use the server controls programmatically.

- ["Steps to Insert a Server Control Programmatically" on page 17](#)
- ["Server Control Properties in Code Behind" on page 20](#)
- ["Customizing Server Controls" on page 31](#)

Steps to Insert a Server Control Programmatically

To insert an Ektron server control programmatically, follow these steps.

["Step 1: Declare the Namespace" on page 17](#)

["Step 2: Create an Instance of the New Control" on page 18](#)

["Step 3: Set the Properties" on page 18](#)

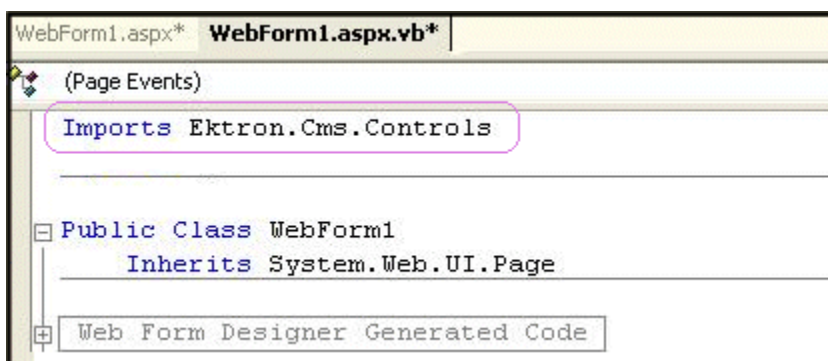
["Step 4: Call the Fill Method" on page 19](#)

["Step 5: Use .text Property to Display the Control" on page 19](#)

Each step is described below. These steps show an example of using the Collection Server Control.

Step 1: Declare the Namespace

Before inserting the server control, you can declare the namespace at the top of the code behind Visual Basic file, as shown below.



NOTE

You do not need to declare a namespace. However if you do not, you must fully qualify objects that you create.

Best Practice

For any customization of the CMS, classes or controls that inherit from Ektron classes, it is recommend that you create your own namespace within 'Ektron.Cms.Custom'. For example, if your company is 'AcmeExampleTech, Inc.' you should create all of your custom classes within the namespace 'Ektron.Cms.Custom.AcmeExampleTech'.

Step 2: Create an Instance of the New Control

Next, declare a control as an object in the code behind. In this example, we create a collection named `MyColl`.

```
dim MyColl as New Collection
```

or

```
dim MyColl as new Ektron.Cms.Controls.Collection
```

With C#, use this syntax.

```
Ektron.Cms.Controls.Collection MyColl = new  
Ektron.Cms.Controls.Collection();
```

You can declare any server control as an object by using the server control name in the ["List of Server Controls" on page 46](#). Another example would be: `Dim MyMdl as New MetaDataList`.

Step 3: Set the Properties**NOTE**

For descriptions of the properties and how to use them, see ["Server Control Properties in Code Behind" on page 20](#).

After inserting the control, you can set the properties that you want to display on the page. For example:

```
Dim MyColl as New Collection  
MyColl.DefaultCollectionID = 4
```

```
MyColl.Page = Page
```

or if you are using code behind to insert the control,

```
dim MyColl as new Ektron.Cms.Controls.Collection  
MyColl.DefaultCollectionID = 4
```

```
MyColl.Page = Page
```

With C#, use this syntax.

```
Ektron.Cms.Controls.Collection MyColl = new  
Ektron.Cms.Controls.Collection();  
MyColl.DefaultCollectionID = 4;
```

```
MyColl.Page = Page;
```

These lines tell the page to display CollectionID 1 unless otherwise specified.

IMPORTANT! When using code behind to add a server control to your Web form, you must set the Page object for the server control to Page. For example, `MyColl.Page = Page`. This line needs to appear between Dim new server control line and the `Fill()` line. This line is not added when dragging and dropping a server control on a Web form. See Also: "Referencing the Page Property" on page 30.

NOTE If you do not know an object's ID number, you can switch to Design mode, drag and drop the object, then use the CMS Explorer to find the ID number. (See "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.) If you do this, remember to delete the dropped object when you are done.

You can also obtain the ID number via the Workarea.

```
Dim MyColl as New Collection
MyColl.DefaultCollectionID = 4
MyColl.Page = Page
MyColl.Random = True
```

This line sets the Random property to **true**. See Also: "Random" on page 183.

Step 4: Call the Fill Method

Because there is no render event when using objects as components not as controls, use the Fill method to fill an object's properties on the page. An example is below.

```
Dim MyColl as New Collection
MyColl.DefaultCollectionID = 4
MyColl.Page = Page
MyColl.Random = True
MyColl.Fill()
```

Step 5: Use .text Property to Display the Control

Finally, use the `.text` property to determine what appears on the Web page. For example, to display the first item in a collection, use this syntax.

NOTE Before adding this line you need to drag and drop a label on your Web form.

```
Dim MyColl as New Collection
MyColl.DefaultCollectionID = 4
MyColl.Page = Page
MyColl.Random = True
MyColl.Fill()
Label1.Text = myColl.EkItems(0).Title
```

To display *all* items in a collection, use this syntax.

```
Dim myColl As New Ektron.Cms.Controls.Collection
Dim ekitem As New Ektron.Cms.Common.ContentBase
myColl.DefaultCollectionID = 2
myColl.Page = Page
myColl.Fill()
Label1.Text = "<ul>"
For Each ekitem In myColl.EkItems
```

```
        Label1.Text &= "<li><a href="" & ekitem.QuickLink &  
"">" & ekitem.Title & "</a>"  
        Next  
        Label1.Text &= "</ul>"
```

This example displays the quick link for every content block in the collection, formatted as a bulleted list. You can use similar code to display a List Summary or search results. The following explains the new (red) code above.

- `label1.Text = ""` displays the opening tag for the bulleted list
- `For Each ekitem In myColl.EkItems` creates a loop for all content blocks in the collection
- `label1.Text &= "" & ekitem.Title & ""` for each content block in the collection, displays its quicklink and title
- `Next` loops through all content blocks in the collection
- `label1.Text &= ""` closes the bulleted list

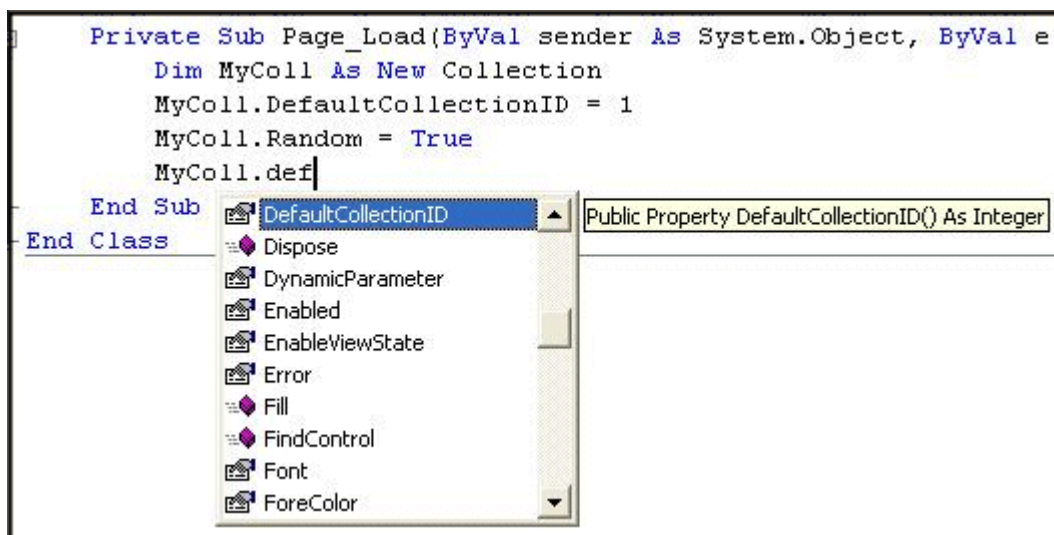
Server Control Properties in Code Behind

Every server control has properties associated with it that you can only access programmatically. This subsection explains, what they are, how to access them, and how to use them.

- ["Displaying Properties for a Control" on page 20](#)
- ["Mutual Server Control Properties in Code Behind" on page 21](#)
- ["Accessing Additional Properties" on page 23](#)
- ["Accessing Items in an Array" on page 30](#)
- ["Referencing the Page Property" on page 30](#)

Displaying Properties for a Control

You can use Visual Studio 2005's intellisense feature to display a control's properties. The intellisense box appears as soon as you insert the period (.) after the object, as illustrated below.



The intellisense box displays *all* properties that can be applied. For a description of Ektron's standard server control properties, see ["List of Server Controls" on page 46](#). To learn about native VS 2005 properties, see its documentation.

The property's tooltip text indicates its type. In the above example, you can see that the DefaultCollectionID's type is integer.

For more information about accessing **Ektron CMS400.NET** object properties, see ["Customizing the Server Control in the Code Behind" on page 32](#).

Mutual Server Control Properties in Code Behind

There are several read-only properties that can be called in the code behind and used by every server control. The list below explains these properties.

NOTE The properties do not display values within Visual Studio.NET during design time. Instead, they only display values at run time, which are dependent on the user's login status.

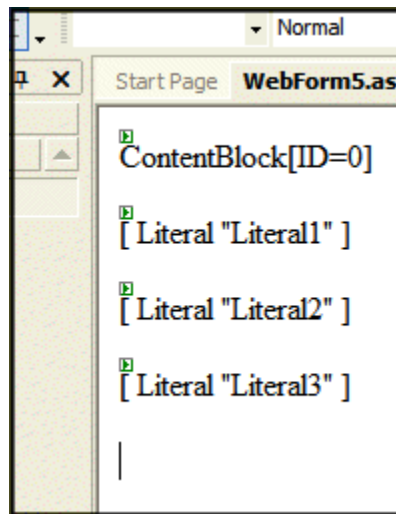
Property	Value	Data Type
IsLoggedIn	Tells if a user is logged in to Ektron CMS400.NET . True = User is logged in False = User is not logged in	Boolean
LoggedInUserName	Gets the Ektron CMS400.NET user name to display.	String

Property	Value	Data Type
LoggedInUserID	Gets the Ektron CMS400.NET ID of the user to display.	Integer

These properties will allow you to personalize any page with your users names and IDs, and show if they are logged in. Here is an example of using these properties in code behind.

NOTE You must be logged in to **Ektron CMS400.NET** for this example to show your name and ID.

1. Drag an **Ektron CMS400.NET** server control onto a Web form.
2. Drag three Literals onto the Web form.

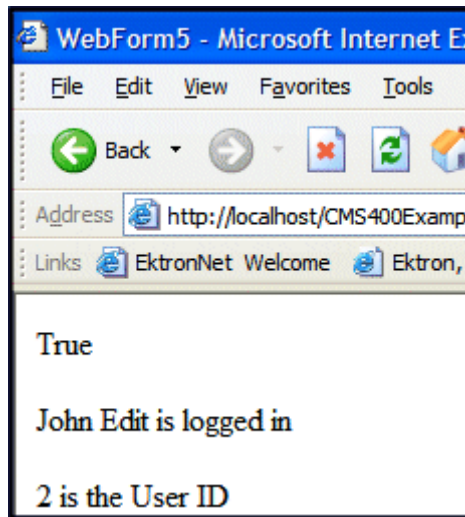


3. Open the code behind for the Web form.
4. Add the following code to the Page_Load event.

```
Literal1.Text = ContentBlock1.IsLoggedIn
If ContentBlock1.IsLoggedIn Then
    Literal2.Text = ContentBlock1.loggedInUserName & " is logged in "
    Literal3.Text = ContentBlock1.loggedInUserID & " is the User ID "
End If
```

5. Build and run the solution.
6. Browse to the login page and log in.
7. Browse to the new Web form you added.

8. The login information is displayed.



Accessing Additional Properties

In addition to the standard properties, **Ektron CMS400.NET** provides access to additional properties for the following objects.

- ListSummary
- Collection
- Search
- ContentBlock
- FormBlock

To access additional properties, use the same syntax you use for standard properties but add `.ekitem` or `.ekitems` after the object. Here is an example.

```
dim MyCB as New ContentBlock
MyCB.DefaultContentID = 30
MyCB.Page = Page
MyCB.Fill()
labell.text = MyCB.EkItem.dateCreated
```

or

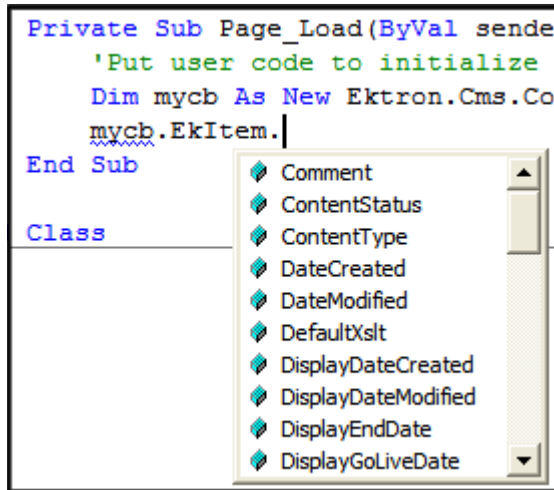
```
dim MyCB as new Ektron.Cms.Controls.ContentBlock
MyCB.DefaultContentID = 30
MyCB.Page = Page
MyCB.Fill()
labell.text = MyCB.EkItem.dateCreated
```

IMPORTANT!

To access additional properties for the Collection, ListSummary, and Search objects, use `ekitems`, not `ekitem`. For example:

`MyColl.ekitems(0).dateCreated`, where (0) is the index of the array. For more Information on using `ekitems`, see "Accessing Items in an Array" on page 30.

You can use intellisense to select from a list of additional object properties, as shown below.



The additional properties are listed below.

NOTE The following properties are read-only. For example, you can get a content block's ID and pass it through to another part of the code, however you cannot set a content blocks ID to be shown. `mycb.Ekitem.id = 8` will not set a content block's ID. The correct way to set a content blocks ID is `DefaultContentID = 8`.

Property	Description	For more information, see
Comment	The content block's comment	

Property	Description	For more information, see
ContentStatus	<p>The status of the content block.</p> <ul style="list-style-type: none"> • approved • checked out • checked in • expired • pending deletion • pending expiration • pending start date • submitted 	Ektron CMS400.NET User Manual, “Appendix A: Content Statuses.”
ContentType	<p>One of the following</p> <ul style="list-style-type: none"> • all types • archived content • archived forms • content • forms 	To learn about archived content, see the Ektron CMS400.NET User Manual, “Scheduling Content to Begin and End” > “Setting Archive Options.”
DateCreated	The date when the content block was created, formatted as a .NET date type	
DateModified	The date when the content block was modified, formatted as a .NET date type	
DefaultXslt	The default Xslt	
DisplayDateCreated	The date when the content block was created. It is formatted as a string that represents Ektron CMS400's display of the date.	
DisplayDateModified	The date when the content block was edited. It is formatted as a string that represents Ektron CMS400's display of the date.	

Property	Description	For more information, see
ContentStatus	<p>The status of the content block.</p> <ul style="list-style-type: none"> • approved • checked out • checked in • expired • pending deletion • pending expiration • pending start date • submitted 	Ektron CMS400.NET User Manual, “Appendix A: Content Statuses.”
ContentType	<p>One of the following</p> <ul style="list-style-type: none"> • all types • archived content • archived forms • content • forms 	To learn about archived content, see the Ektron CMS400.NET User Manual, “Scheduling Content to Begin and End” > “Setting Archive Options.”
DateCreated	The date when the content block was created, formatted as a .NET date type	
DateModified	The date when the content block was modified, formatted as a .NET date type	
DefaultXslt	The default Xslt	
DisplayDateCreated	The date when the content block was created. It is formatted as a string that represents Ektron CMS400’s display of the date.	
DisplayDateModified	The date when the content block was edited. It is formatted as a string that represents Ektron CMS400’s display of the date.	

Property	Description	For more information, see
DisplayEndDate	The content block's end date. It is formatted as a string that represents Ektron CMS400's display of the date.	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting an End Date on a Content Block"
DisplayGoLiveDate	The content block's start date. It is formatted as a string that represents Ektron CMS400's display of the date.	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting a Start Date"
DisplayStartDate	The content block's start date. It is formatted as a string that represents Ektron CMS400's display of the date.	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting a Start Date"
EndDate	The content block's end date, formatted as a .NET date type	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting an End Date on a Content Block"
EndDateAction	<ul style="list-style-type: none"> archive display archive expire refresh report 	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting Archive Options."
FolderID	The ID of the folder that contains each content block	
GoLiveDate	The content block's start date formatted as a .NET date type	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting a Start Date"
Html	The content that makes up the content block. If content block is in XML it will return it as raw XML content	
Hyperlink	Content block title wrapped by <a href> tags	
Id	The content block 's ID number	
InheritedFrom	If folder permissions are inherited, the folder from which they are inherited	Ektron CMS400.NET Administrator manual "Setting Permissions" > "Inheritance"

Property	Description	For more information, see
IsInherited	Whether a content block's permissions are inherited	Ektron CMS400.NET Administrator manual "Setting Permissions" > "Inheritance"
IsPrivate	Whether or not a content block is private	Ektron CMS400.NET Administrator manual "Managing Users & User Groups" > "Private Content & Memberships"
Language	The content block's language	Ektron CMS400.NET Administrator manual "Multi-Language Support" > "Working with Multi-Language Content"
LastEditorFname	The first name of the last person to edit the content block	
LastEditorLname	The last name of the last person to edit the content block	
PackageDisplayXSLT	If the content block is XML, the name of its XSLT	
QuickLink	The content block's quicklink	Ektron CMS400.NET User manual "Library Folder" > "Quicklinks"
StartDate	The content block's start date formatted as a .NET date type	Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" > "Setting a Start Date"

Property	Description	For more information, see
Status	<p>The status of the content block</p> <ul style="list-style-type: none"> • approved • checked out • checked in • expired • pending deletion • pending expiration • pending start date • submitted 	Ektron CMS400.NET User manual “Appendix A: Content Statuses”
Teaser	The content block summary	“Adding a Content Block Summary” in the Ektron CMS400.NET User manual
TemplateLink	Currently empty and not being used with the ContentBlock server control	
Title	The content block title	
UserID	Last user who edited the content	
Xslt1	The content block’s first Xslt, as defined in Ektron CMS400.NET	
Xslt2	The content block’s second Xslt, as defined in Ektron CMS400.NET	
Xslt3	The content block’s third Xslt, as defined in Ektron CMS400.NET	
Xslt4	The developer can use this property programmatically. Ektron CMS400.NET only uses Xslt1, 2 and 3 in the workarea.	
Xslt5	The developer can use this property programmatically. Ektron CMS400.NET only uses Xslt1, 2 and 3 in the workarea.	

Accessing Items in an Array

To access and manipulate content blocks returned by an object, use the common class `Ektron.Cms.Common.ContentBase`. *EkItems* is an array of `Ektron.Cms.Common.ContentBase`. *EkItem* is a single `Ektron.Cms.Common.ContentBase`.

Search, Collection and ListSummary have *EkItems* (an array of `ContentBase`), while `ContentBlock` has a single *EkItem*. Here is an example of how to use this feature.

```
dim MyC as new Ektron.Cms.Controls.Collection
MyC.DefaultCollectionID = 1
MyC.Page = Page
MyC.Fill()
dim item as Ektron.Cms.Common.ContentBase
MyC.Text = "<ul>"
for each item in MyC.EkItems
    MyC.Text &= "<li>" & item.Title & "</li>"
next
MyC.Text &= "</ul>"
Response.Write(MyC.Text())
```

This example formats every item in the collection in a bulleted list.

Referencing the Page Property

Server controls require a reference to their parent page (for example, utilizing the `DynamicParameter` property on a content block to check for a query string), you must provide access to the page object if you declared your control in the code behind. To do this, set the control's page property to the Web page you're working on. For an example, see the code in red and italics below.

```
dim search as new Ektron.Cms.Controls.Search()
MySearch.Page = Page
MySearch.Fill()
```

With C#, use this syntax.

```
Ektron.Cms.Controls.Search MySearch = new
Ektron.Cms.Controls.Search();
MySearch.Page = Page;
MySearch.Fill();
```

This relationship is only required when inserting a control in the code behind. When dragging and dropping, even if you make changes in code behind, the relationship is automatically generated.

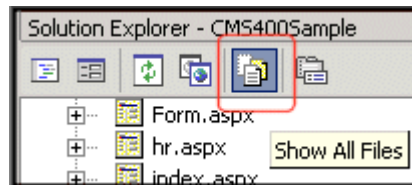
Best Practice

Ektron recommends including the page property reference when using the server controls as components in the code behind.

Customizing Server Controls

One of Visual Studio 2005's strengths is its separation of coding and logic from presentation. Web page formatting is handled by a page's HTML, while the logic is handled by the code behind, which is stored in the corresponding .vb file. For example, if the ASP.NET page is mypage.aspx, the code behind file is mypage.aspx.vb.

NOTE If you do not see the code behind files, click the Show All Files button on the VS 2005 Solution Explorer toolbar.



Within the vb file, you can use Visual Basic to insert code to manipulate the events that occur on the page.

This subsection contains the following topics:

- "Recognizing the Server Control Within the HTML" on page 31
- "Recognizing the Server Control Within the Code Behind" on page 31
- "Customizing the Server Control in the Code Behind" on page 32
- "Troubleshooting Error Creating Control Message" on page 32

Recognizing the Server Control Within the HTML

Within a Web page's HTML, a <cms> tag wraps the VS 2005 object, as shown below.

```
<cms:Search id="Search1" runat="server" ButtonText="Search" Display="Vertical"></cms:Search>
```

Above is an example of a Search Server Control. Below is an example of a ContentBlock Server Control.

```
<cms:ContentBlock id="ctrlMainContentBlock" runat="server" DefaultContentID="1"
DynamicParameter="id" OverrideXslt="Default"></cms:ContentBlock>
```

Recognizing the Server Control Within the Code Behind

Within the VS 2005 code behind file, the Ektron server controls appear (along with the VS 2005 controls) in the Web Form Designer Generated Code section. When you click the plus sign (+) to display this section, you see something like the following. The content block listed in HTML above is circled below to help you see their relationship.

```

#Region " Web Form Designer Generated Code "
' This call is required by the Web Form Designer.
<System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent() ...
Protected WithEvents MetaDataArea As System.Web.UI.WebControls.Literal
Protected WithEvents DhtmlJavaScript As System.Web.UI.WebControls.Literal
Protected WithEvents DropDownMenu As System.Web.UI.WebControls.Literal
Protected WithEvents ctrlMainContentBlock As Ektron.Cms.Controls.ContentBlock
Protected WithEvents ctrlTopContentBlock As Ektron.Cms.Controls.ContentBlock
Protected WithEvents ctrlBottomLeftContentBlock As Ektron.Cms.Controls.ContentBlock
Protected WithEvents ctrlBottomRightContentBlock As Ektron.Cms.Controls.ContentBlock
Protected WithEvents ctrlLanguageSelect As Ektron.Cms.Controls.LanguageSelect
'NOTE: The following placeholder declaration is required by the Web Form Designer. ...
Private designerPlaceholderDeclaration As System.Object

```

The next section of the code behind page loads the page into the browser.

```

Private Sub Page_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load

```

We want our events to occur while the page is loading, so we'll add custom code following this line.

Customizing the Server Control in the Code Behind

To customize an Ektron server control in the code behind, insert code similar to the following *after* the Page_load command.

```
Dim MyObj As New Ektron.Cms.Controls.ContentBlock
```

This code declares a variable named MyObj and assigns to it the value of a content block. The content block is part of the Ektron.CMS.Controls content base, so it has access to the **Ektron CMS400.NET** database.

After defining MyObj as a content block, you can access its properties. For example, to assign a defaultID of 24, insert the following.

```
Dim MyObj As New Ektron.Cms.Controls.ContentBlock
MyObj.DefaultContentID = 24
```

Now, the content block can be specified dynamically in the URL of the hyperlink that calls it. If not, content block 24 displays.

This is just an example of programmatically applying property values to content blocks. For a complete list of properties available to **Ektron CMS400.NET** objects, see "[List of Server Controls](#)" on page 46.

Troubleshooting Error Creating Control Message

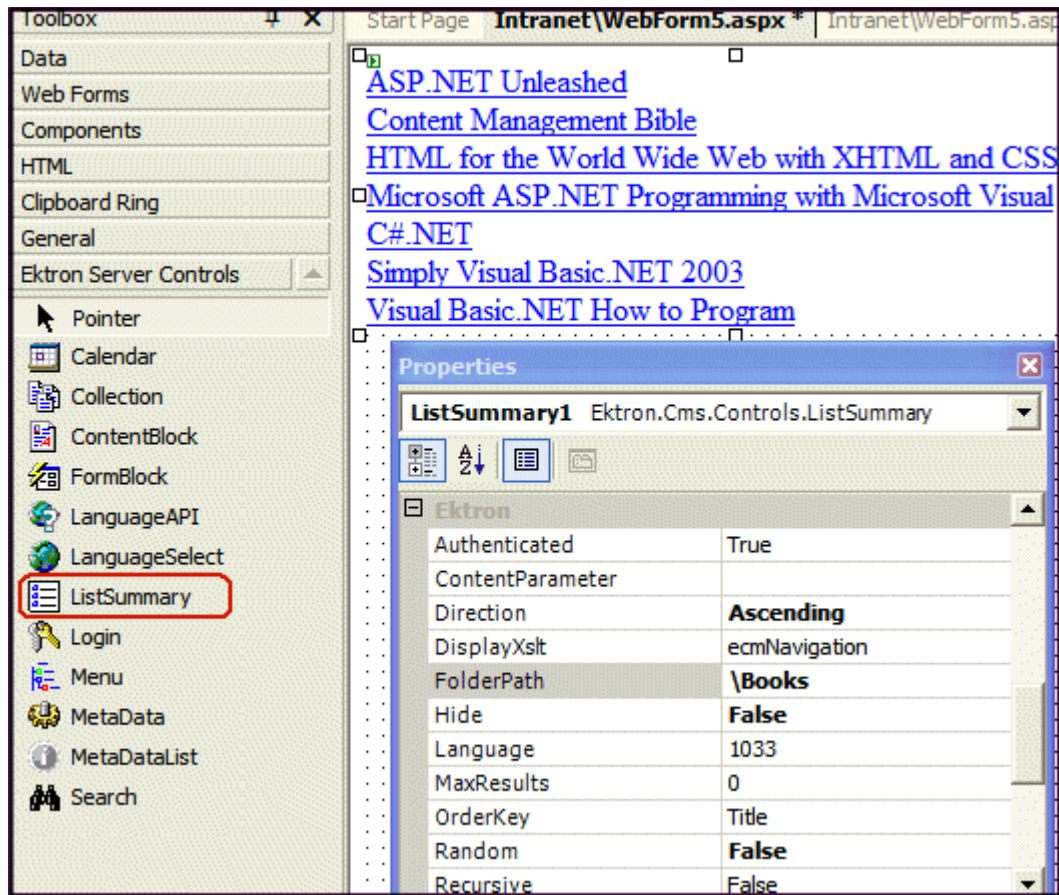
If you get an Error Creating Control message while trying to use a server control, you can view the text of the error message by hovering the mouse over the control. See illustration below.



Using Drag & Drop and Programmatically Together

All server controls can be further manipulated after dragging and dropping them on a web form by using the code behind. Here is an example of using a drag and drop ListSummary Server Control then modifying it programmatically.

First, drag and drop a List Summary on your Web form and set your properties.



Then add the following code to the code behind.

```
Dim myString As String
Dim i
For i = LBound(ListSummary1.EkItems) To UBound(ListSummary1.EkItems)
myString &= "<a href=""" & ListSummary1.EkItems(i).QuickLink & """>" &
ListSummary1.EkItems(i).DateCreated & "</a><br>"
ListSummary1.Text = myString
Next
```

An Explanation of the code:

ListSummary1 is the ID of the object. It is used to get access to its properties.

Create a string that contains the output (myString).

```
Dim myString As String
```

Set the object's text property to that string.

```
myString &= "<a href=""" & ListSummary1.EkItems(i).QuickLink & """>" &
ListSummary1.EkItems(i).DateCreated & "</a><br>"
```

Wrap in a loop so it loops through each of the items.

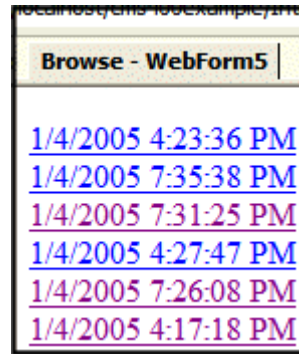

```

Dim i
For i = LBound(ListSummary1.EkItems) To UBound(ListSummary1.EkItems)

Next

```

This example outputs the date created for each content block in a List Summary.



Data Binding with Server Controls

With data binding, you can bind Ektron server controls to a GridView Control, DataList Control, or Repeater Control. This gives more flexibility when you use data from the Ektron server controls. Benefits of data binding include ease of data manipulation and the ability to format data.

Title	Teaser
Ektron Rated Positive	Ektron Inc., an innovator in Web content management software, today announced ...
Ektron to Demonstrate Healthcare	“Healthcare Content Indexing Framework” which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.

IMPORTANT!

When HTML is bound to a column, you need to add `HtmlEncode = False` to it. Otherwise, the HTML appears as code. For example, `<p>Ektron Inc., an innovator in Web content management software, today announced...</p>`.

The following are data bindable Ektron Server Controls.

- Collection
- Content Block
- IndexSearch

- Form Block
- List Summary
- Poll
- RssAggregator
- Site Map
- Search

The following is an example of code behind that uses a GridView to display a Collection:

```
Dim myCol As New Ektron.Cms.Controls.Collection
myCol.DefaultCollectionID = 4
myCol.Page = Page
myCol.Fill()
GridView1.DataSourceID = myCol
GridView1.DataBind()
```

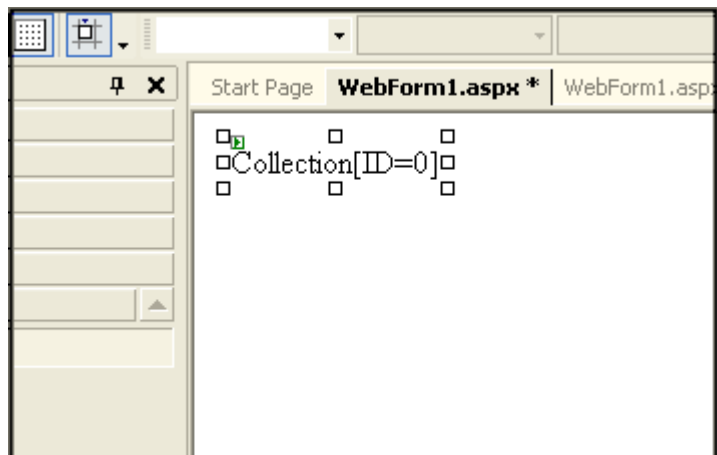
This subsection contains the following topics:

- ["Steps to Data Binding Using Drag and Drop Server Controls" on page 36](#)

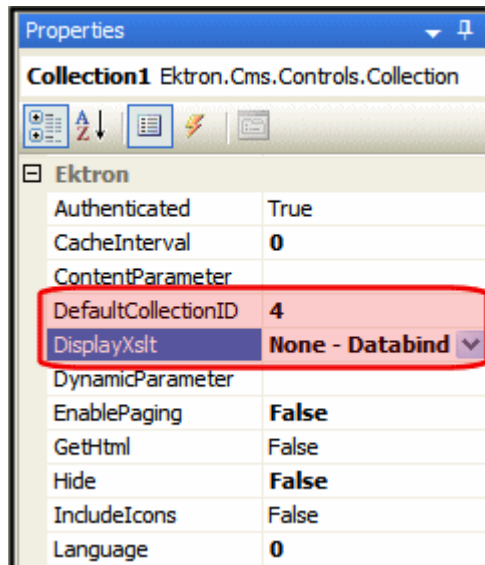
Steps to Data Binding Using Drag and Drop Server Controls

The following example uses a Collection.

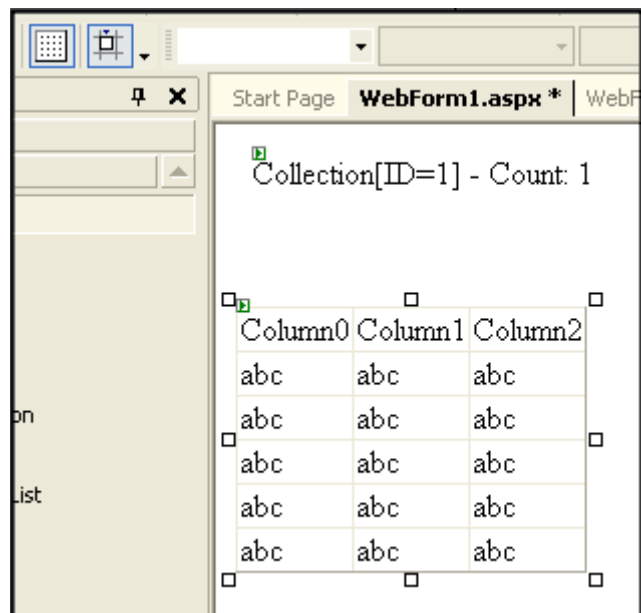
1. Create a new Web form.
2. Drag and drop a data bindable Server Control on the Web form. For example, a Collection server control.



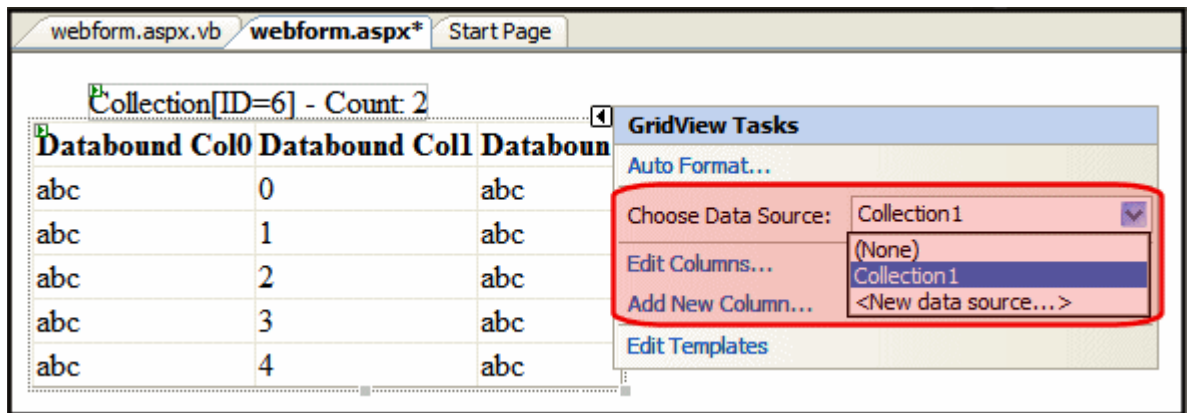
3. In properties, choose a DefaultCollectionID for the Server Control and make sure DisplayXslt is set to **None- DataBind Only**.



4. Drag and drop a GridView on the Web form.



5. In the GridView Tasks, choose the DataSourceID.



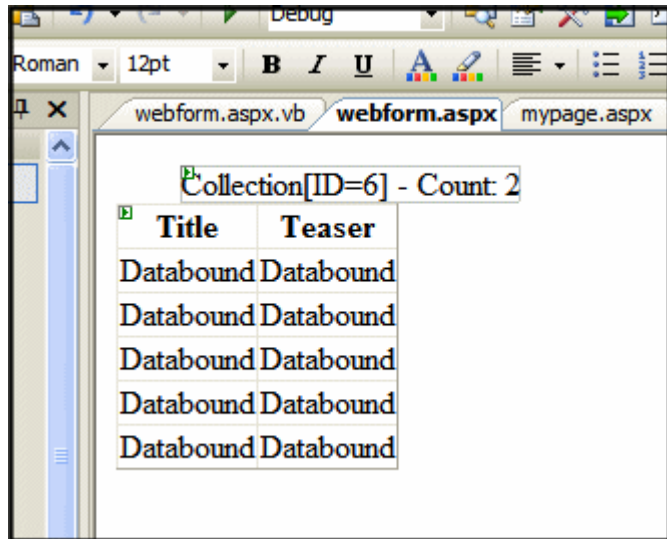
6. In code behind, add the following line of code to the Page init event:
`Collection1.Fill()`
7. Select the columns to be databound by clicking **Add New Column** in GridView Tasks.
8. Complete the Add Field window according to the table below.

Field	Entry
Choose a field type:	BoundField

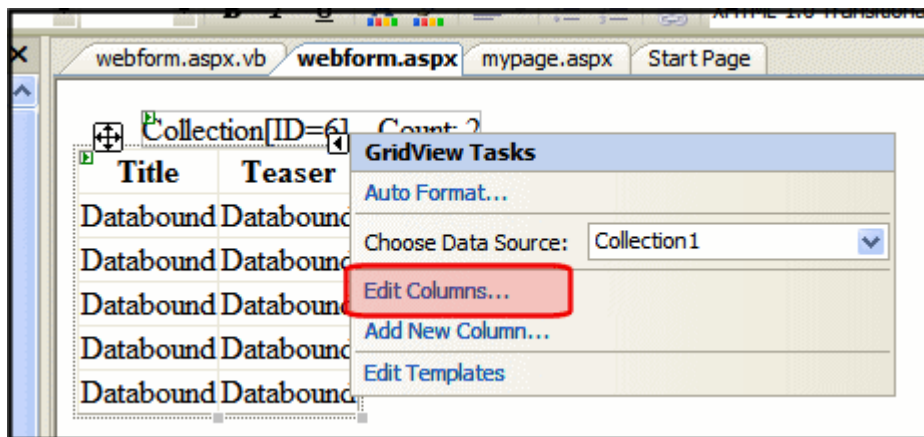
Field	Entry
Header text:	The title name for each column.
Data field:	<p>The data to bind to each column. Choose one of the following:</p> <p>ID - the content block ID</p> <p>Title - the content block title</p> <p>Teaser - the content summary</p> <p>Html - the html content</p> <p>StartDate - the content's start date</p> <p>DateModified - the content's last modified date</p> <p>EndDate - the content's end date</p> <p>LastEditorFname - the last editor's first name</p> <p>LastEditorLname - the last editor's last name</p> <p>QuickLink - the content's quicklink</p> <p>HyperLink - the content's hyperlink</p> <p>DisplayStartDate - the string representation of the start date</p> <p>FolderID - the folder ID where the content is located</p> <p>ContentStatus - the content's status</p> <p>Language - the content's default language</p> <p>DisplayDateModified - string representation of the content's last modified date</p> <p>DisplayEndDate - string representation of the content's end date</p> <p>EndDateAction - an action tied to end date. For example, Refresh_Report</p> <p>Comment - the content's comments</p>

9. Click **OK**.

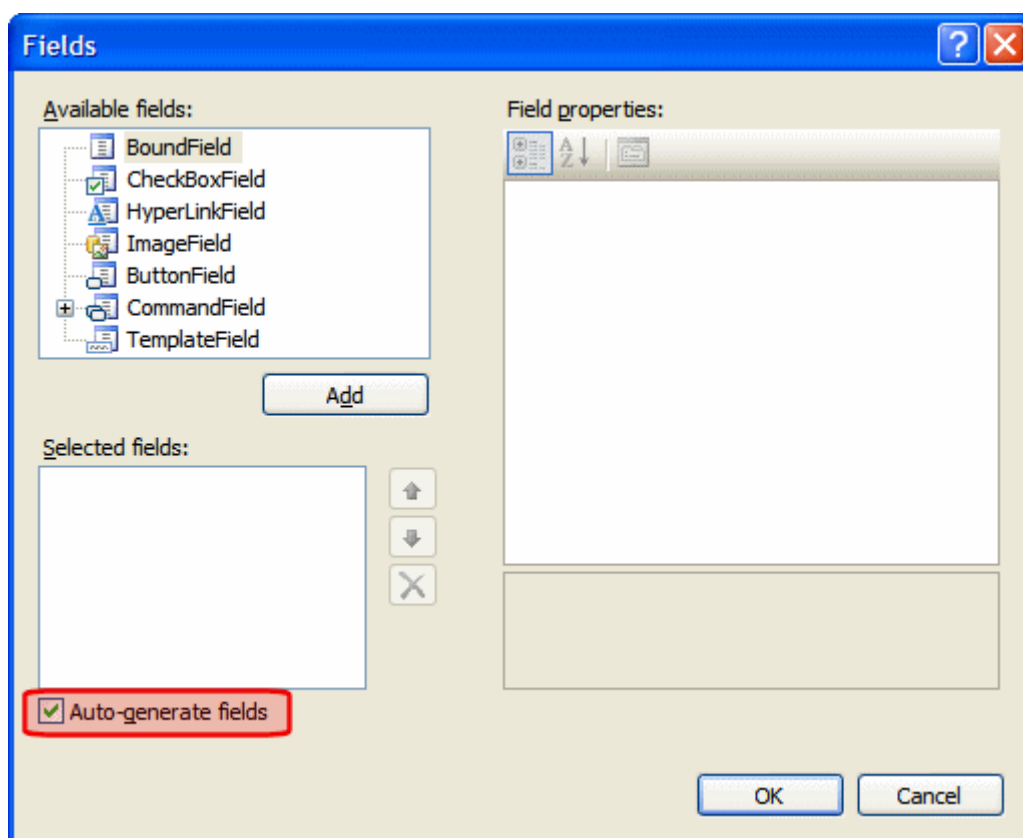
10. Repeat steps six, seven and eight for each column you want to add.



11. If you want to add all of the columns automatically, in the GridView Task menu click Edit Columns. Otherwise, skip to step thirteen.



12. Click the **Auto-generate fields** checkbox.



13. Click **OK**.
 14. From the Build menu, click **Build Page**.
 15. In design view, right click the form and select **View in Browser**.

Title	Teaser
Ektron Rated Positive	Ektron Inc., an innovator in Web content management software, today announced ...
Ektron to Demonstrate Healthcare	"Healthcare Content Indexing Framework" which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.

For more information on GridView, DataList, Repeater and DetailsView, see the help inside Visual Studio.

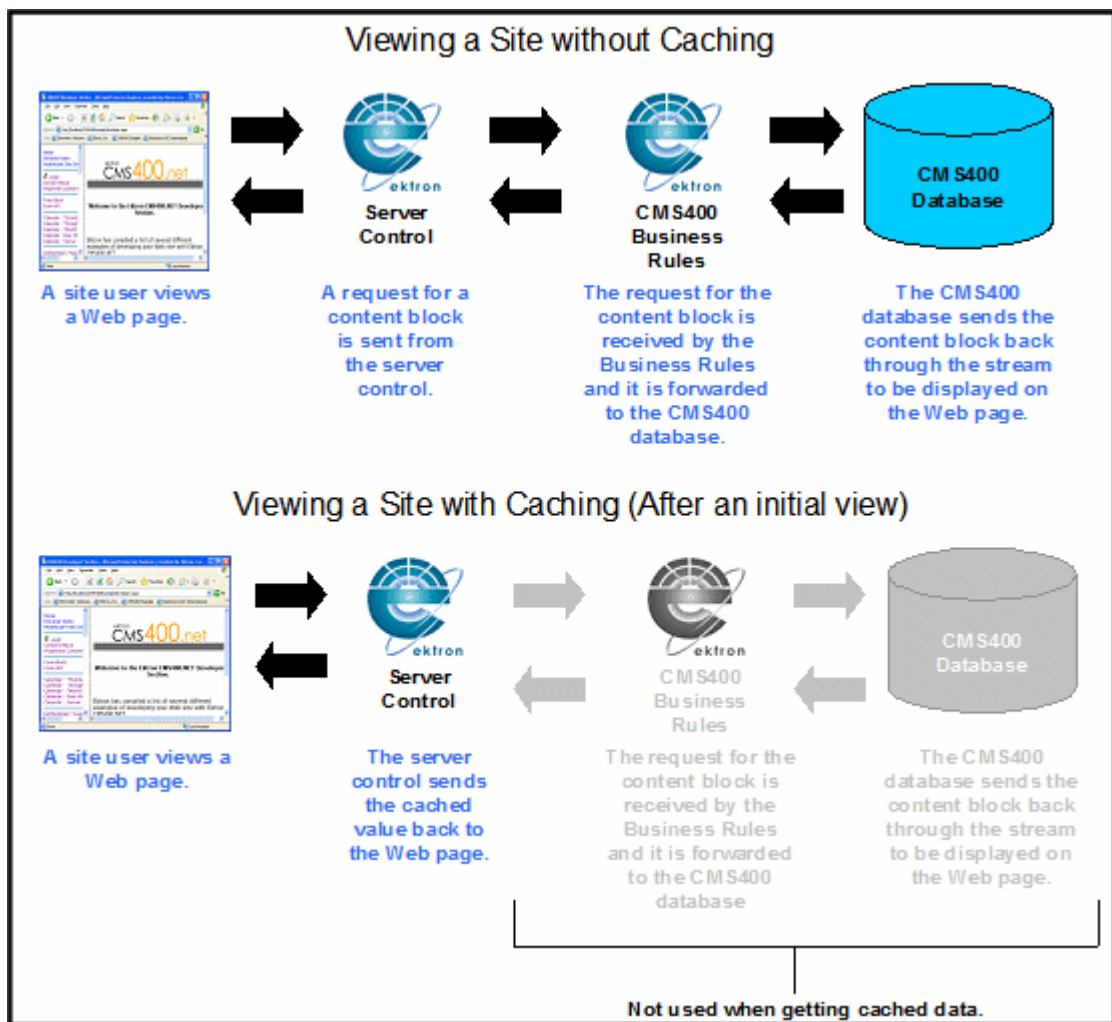
Caching with Server Controls

A key factor in building high-performance, scalable Web applications is the storage of items in memory after the first time they are requested. The items include data objects, pages, and parts of a page.

Known as caching, this process saves and later reuses page output or application data across HTTP requests. You can store items on the Web server or other software in the request stream, such as the proxy server or browser.

Caching saves time and resources because the server does not have to recreate information, particularly things that demand significant processor time or other resources.

The following illustrates data flow in non-cached and cached environments.



Ektron CMS400.NET provides two kinds of caching.

- **Caching Individual Content** - lets you cache part of a Web page; available with some server controls
- **Page Level Caching** - lets you cache an entire page; available with all server controls

Each is explained below.

Caching Individual Content

There are two kinds of caching of individual content.

- "Caching While not Logged In" on page 43
- "Caching While Logged In" on page 43

Caching While not Logged In

For a description of how Caching While not Logged In works, see "Caching with Server Controls" on page 42.

Caching While Logged In

When a user is logged in, caching prevents any changes from appearing on the site for the number of seconds defined in the cache interval. This reduces the number of database hits, which improves your server's performance.

For example, if you add a new item to a Collection, the change only appears on the site when the cache interval expires. However, the user can switch to Preview mode to see the updated Collection immediately. See Also: **Ektron CMS400.NET User Manual** > "Introduction to Ektron CMS400.NET" > "Logging In and Out" > "Site Preview."

IMPORTANT!

Caching while logged in does not work with Private content. See Also: "Private Content & Memberships" section of the "Managing Users & User Groups" chapter of the Ektron CMS400.NET Administrator Manual

Setting up Caching of Individual Content

To set up caching of individual content for any Web page, follow these steps.

1. Make sure the Web.config file's `ek_CacheControls` property's value = "1" (one). This is the default. This setting lets you control caching with server controls.
2. Add a server control that supports caching of individual content to a Web form. See "The caching of individual content is available for the server controls listed below." on page 44.

3. Set the server control's `CacheInterval` property to the length of time the data should be cached. For example, to cache data for five minutes, set `CacheInterval` to 300 (three hundred).

The default value is 0 seconds. So, you must change the default to enable caching.

Server Controls that Support Caching of Individual Content

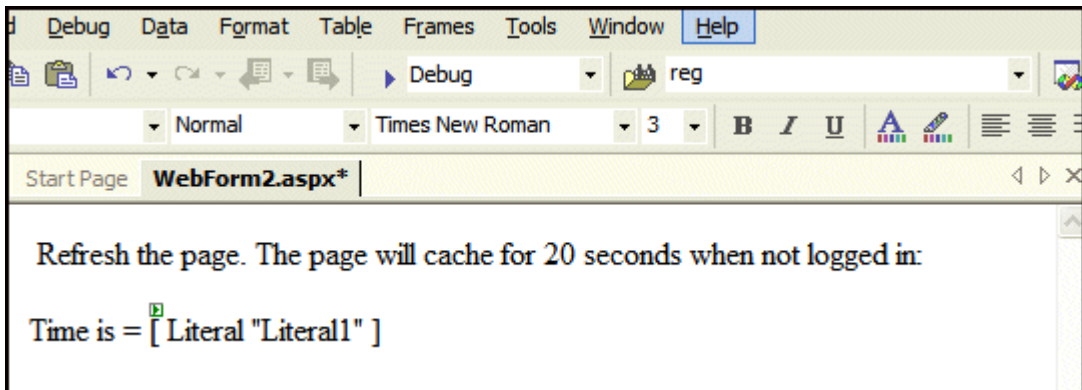
The caching of individual content is available for the server controls listed below.

Control	Caching available while not logged in	Caching available while logged in
ActiveTopics	X	
All blog controls	X	
Collections	X	X
ContentBlock	X	
ContentList	X	X
DhtmlMenu	X	X
FormBlock	X	
Forum	X	
ListSummary	X	X
MetadataList	X	X
Menu	X	X
Poll	X	
PostHistory	X	
SiteMap	X	
SmartMenu	X	X

Page Level Caching

Below is an example of a page level cache for non logged-in users. In this example, you use a Server Control to define whether a user is logged in or not. Then, define if the time is cached, based on the user's status.

1. Create a new Web form in your Ektron CMS400.NET project.
2. Add the following text to the Web form:
Refresh the page. The page will cache for 20 seconds when not logged in:
Time is =
3. Next to **Time is =**, add a literal.



4. Below that, add a ContentBlock Server Control.
5. Set the DefaultContentID. For this example, DefaultContentID = 20.
6. Add the following to the code behind.

```
If Not ContentBlock1.IsLoggedIn Then
    Response.Cache.SetExpires(DateTime.Now.AddSeconds(20))
    Response.Cache.SetCacheability(HttpCacheability.Public)
    Response.Cache.SetValidUntilExpires(True)
    Response.Cache.VaryByParams("id") = True
    Response.Cache.SetVaryByCustom("cmsCache")
End If
Literal1.Text = Now()
```

7. Build your Web form.
8. Browse to your Web form using your browser.
9. Press your browser's refresh button.

If you are not logged in to CMS400.NET, the time remains for twenty seconds. After twenty seconds, when you refresh, the new time appears.

NOTE You can use the same code in a user control to cache output in a particular region of the page.

Using Ajax Enabled Server Controls and Custom Code

Best Practice

When using Ajax server controls and custom code, wrap the custom code in a check for “is not a callback” so it is not executed when a callback from an Ajax server control occurs.

If you are use an Ajax-enabled server control and write custom code, it may generate an exception, indicating the server control does not work. This issue happens during callback for the Ajax server control when custom code accesses a property that is filled during page load, but not filled during callback.

For example, a Poll server control is on a Web form, and you want the title of a content block to appear in a literal on that form. The EkItem Title property for the content block is filled upon page load. When a site user answers the poll question, an exception occurs during the callback because the EkItem Title property is not refilled. However, the site user does not see the exception. It just looks like the Poll server control is not working.

Below is an example of custom code that makes the content block’s title appear in the literal:

```
Literall1.Text = ContentBlock1.EkItem.Title
```

To solve this issue, wrap the custom code in a check for “is not a callback”. This prevents the code execution when callback occurs. For example,

```
[C#]
If( !IsCallback ) {
Literall1.Text = ContentBlock1.EkItem.Title
}
```

```
[VB]
If ( Not IsCallback )
Literall1.Text = ContentBlock1.EkItem.Title
End If
```

List of Server Controls

To customize your Ektron CMS400.NET application, use these server controls.

Server Control Names	Description	More Information
AnalyticsTracker	Track statistics about visits to your Web site.	"Content Analytics Server Control" on page 106

Server Control Names	Description	More Information
AssetControl	The AssetControl server control, when viewed on a Web form, displays a drag and drop box for users to upload assets to CMS400.NET or update an existing asset.	"AssetControl Server Control" on page 51
Blog	<p>Ektron provides nine server controls that relate to displaying a Blog on a Web site. They are:</p> <ul style="list-style-type: none"> • The Blog server control • The BlogCalendar server control • The BlogCategories server control • The BlogEntries server control • The BlogPost server control • The BlogRecentPost server control • The BlogRoll server control • The BlogRSS server control • The BlogArchive server control 	"Blog Server Controls" on page 54 "Blog Server Control" on page 54 "BlogEntries Server Control" on page 59 "BlogPost Server Control" on page 61 "BlogCalendar Server Control" on page 63 "BlogRoll Server Control" on page 64 "BlogCategories Server Control" on page 65 "BlogRecentPosts Server Control" on page 67 "BlogRSS Server Control" on page 68 "BlogArchive Server Control" on page 70
BreadCrumb	Creates a breadcrumb trail of where site visitors have been and allows them to navigate back to previous pages.	"BreadCrumb Server Control" on page 72 For information on other Breadcrumb types, see "FolderBreadcrumb Server Control" on page 82 and "SiteMap Server Control" on page 368
BusinessRules	Adds a Business Rule created in the CMS400.NET Workarea to a Web form.	"BusinessRules Server Control" on page 86
Calendar	Displays an event calendar.	"Calendar Server Control" on page 88
Collection	Displays links generated by collections.	"Collection Server Control" on page 95
ContentBlock	Displays and manages a content block.	"ContentBlock Server Control" on page 108
ContentList	Displays a list of content blocks created from a comma delimited list.	"ContentList Server Control" on page 117

Server Control Names	Description	More Information
ContentRating	Place a rating scale on any page of your Web site. The scale lets Ektron CMS400.NET collect feedback by giving site visitors the opportunity to rate a particular content item.	"ContentRating Server Control" on page 134
DesignTimeDiagnostic	The DesignTimeDiagnostic server control is used to verify the connection to Ektron's server controls Web service.	"DesignTimeDiagnostic Server Control" on page 139
DhtmlMenu	Displays a drop down menu with submenus.	"DhtmlMenu Server Control" on page 254
Directory	The Directory server control lets you customize the behavior of the Taxonomy feature. You place this control on any Web form to display a taxonomy.	"Directory Server Control" on page 141
FolderBreadcrumb	FolderBreadcrumbs show you the path through the sitemap to the current page.	"FolderBreadcrumb Server Control" on page 82 For information on other Breadcrumb types, see "Breadcrumb Server Control" on page 72 and "SiteMap Server Control" on page 368
FormBlock	Displays a content block that is associated with a form.	"FormBlock Server Control" on page 159
Forum	Forums provide a discussion board where topics can be discussed on your Web site. There are three server controls associated with discussion boards. They are: <ul style="list-style-type: none"> • The Forum server control • The ActiveTopics server control • The PostHistory server control 	"Discussion Board Server Controls" on page 147 "Forum Server Control" on page 147 "ActiveTopics Server Control" on page 155 "PostHistory Server Control" on page 157
HTMLEditor	The server control displays the Ektron CMS400 HTML editor on a Web page. By adding the HTMLEditor to a Web page, site visitors can create or edit HTML content for a site or project.	"HTMLEditor Server Control" on page 163

Server Control Names	Description	More Information
ImageControl	Display a CMS400.NET image on a Web page. In addition, if a user has permission to edit the image, he can right click the image and a drag and drop box appears. This box allows a user to overwrite the existing file.	"ImageControl Server Control" on page 171
IndexSearch	Perform XML index searches on XML content. (XML content is created using Smart Forms.)	"IndexSearch Server Control" on page 351 For information on searching other content types, see "Search Server Control" on page 318
LanguageAPI	Allows developers to force a language for a Web site.	"LanguageAPI Server Control" on page 173
LanguageSelect	Lets user select language of site.	"LanguageSelect Server Control" on page 176
ListSummary	Displays a list of content block summaries.	"ListSummary Server Control" on page 178
Login	Displays the login or logout button, which lets the user log in and use Ektron CMS400.NET.	"Login Server Control" on page 186
Map	Displays a map that flags locations of interest to your site visitors. Each location is a CMS content item to which map information was added.	"Map Server Control" on page 190
Membership	Creates a form for site visitors to add themselves to your Membership users list.	"Membership Server Control" on page 214
Menu	Calls a defined menu to be used with an Xslt to display the menu.	"Menu Server Control" on page 246
MetaData	Adds metadata contained in content blocks to a Web page.	"MetaData Server Control" on page 261

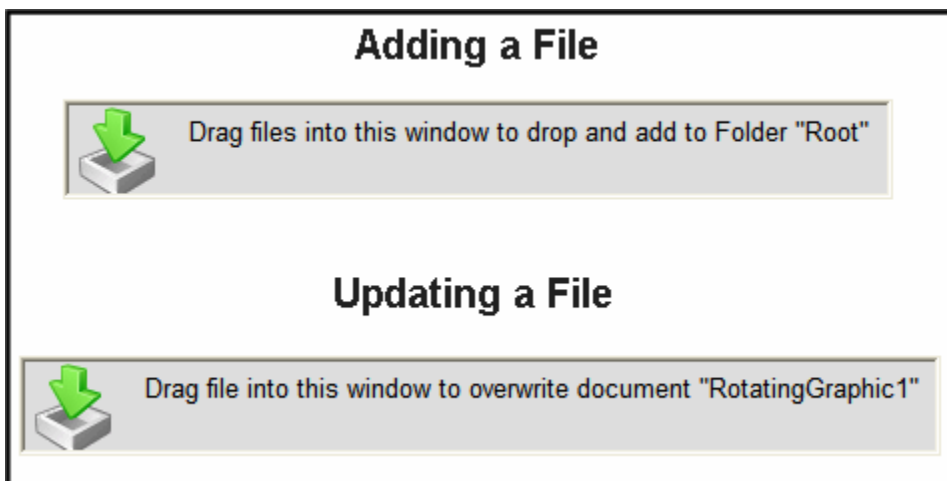
Server Control Names	Description	More Information
MetaDataList	Creates a list of content blocks based on the metadata contained in the content blocks.	"MetaDataList Server Control" on page 266
Personalization	<p>Personalization server controls allow you to create a Web page that site users can personalize. The following server controls allow for personalization:</p> <ul style="list-style-type: none"> • EktronCatalogPart • EktronCommunityCatalogPart • EktronWebPartZone • PersonalizationManager 	"EktronCatalogPart Server Control" on page 279 "EktronCommunityCatalogPart Server Control" on page 282 "EktronWebPartZone Server Control" on page 283 "PersonalizationManager Server Control" on page 284
Poll	Displays a poll or survey created from an Ektron CMS400.NET form on a Web page.	"Poll Server Control" on page 298
RssAggregator	Aggregates an RSS feed from an RSS Web site.	"RssAggregator Server Control" on page 302
Search	Inserts a text box with a search button in the template. The site visitor enters criteria and launches a search of HTML content, forms, and assets on your Web site.	"Search Server Control" on page 318 For information on using XML Indexing search, see "IndexSearch Server Control" on page 351
Sitemap	The Sitemap server control utilizes the folder breadcrumb information in the Workarea to display a sitemap of your site.	"SiteMap Server Control" on page 368 For information on other Breadcrumb types, see "Breadcrumb Server Control" on page 72 and "FolderBreadcrumb Server Control" on page 82
SmartMenu	The Smart Menu server control displays a menu on a Web page. This menu style is the most robust of the three delivered with Ektron CMS400.NET .	"Smart Menu Server Control" on page 226
WebSearch	The Web Search server control lets you customize the behavior of the search.	"Web Search Server Control" on page 306

AssetControl Server Control

The AssetControl server control, when viewed on a Web form, displays a drag and drop box for users to upload assets to CMS400.NET or update an existing asset. This box is similar to the upload box in the Workarea. See *Also*: The User Manual section “Using the Document Management Functionality” > “Importing Files via Drag and Drop”.

The difference between the Workarea and the server control is, in the Workarea users can only upload assets. With the AssetControl server control, users can upload assets or update an existing asset by overwriting it. Even though the asset is overwritten, the previous version of the asset is still available through the history feature in CMS400.NET. See *Also*: The User Manual section “Viewing and Restoring Previous Content”.

The UploadType property allows you to switch the control between uploading assets and updating them. Note in the image below, when the property is set to Add, the message in the drag and drop box states to which folder the assets are uploaded. When the property is set Update, the box states which asset is updated.



This section contains the following topics:

- ["AssetControl Server Control Properties" on page 51](#)

AssetControl Server Control Properties

The AssetControl server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
DefaultAssetID	The ID of the asset you want to update. This property is used when the UploadType property is set to Update . See Also: "UploadType" on page 53 If you don't know the ID number of the asset, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DefaultFolderID	The ID of the folder where assets are added. This property is used when the UploadType property is set to Add . See Also: "UploadType" on page 53 If you don't know the ID number of the folder, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DynamicParameter	Gets or sets the Querystring parameter to read a content ID or folder ID dynamically. The content ID is read when the UploadType property is set to Update. The folder ID is read when UploadType property is set to Add. To use the default content ID or default folder ID, leave blank. See Also: "UploadType" on page 53	String
Hide	Used to hide output of AssetControl in design time and run time. True = Hide AssetControl False = Display AssetControl	Boolean
Language	Set a language for viewing the collection. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer

Property	Description	Data Type
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
UploadType	<p>Select whether the control adds new assets or updates existing ones.</p> <ul style="list-style-type: none"> Select Add to add assets and use the DefaultFolderID property. See Also: "DefaultFolderID" on page 52. <p>If a file of the same name already exists in the folder, the new file is created using the naming convention <code>filename(2)</code>.</p> <ul style="list-style-type: none"> Select Update to update assets. In this case, you <i>must</i> identify an asset at the DefaultAssetID property. See Also: "DefaultAssetID" on page 52. <p>The default is Add.</p>	String
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is Span.</p> <p>Span - The tag is used to designate an inline portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

Blog Server Controls

IMPORTANT!

In addition to reading this section on the Blog server controls, reading the User Manual section “Blogs” is recommend.

Ektron provides nine server controls that relate to displaying a Blog on a Web site. One of these, the Blog server control, allows you to display all of the pieces that typically make up a blog site on a Web form. These items are blog entries, blog posts, blog categories, a blog's recent posts, a blog's RSS feed, an archive, a blog calendar and a blog roll. The other seven Blog server controls allow you to display each of these items individually. The eight individual Blog server controls offer additional ways to customize the page layout and appearance of the items.

The nine Blog server controls provided by Ektron are

- “Blog Server Control” on page 54
- “BlogEntries Server Control” on page 59
- “BlogPost Server Control” on page 61
- “BlogCalendar Server Control” on page 63
- “BlogRoll Server Control” on page 64
- “BlogCategories Server Control” on page 65
- “BlogRecentPosts Server Control” on page 67
- “BlogRSS Server Control” on page 68
- “BlogArchive Server Control” on page 70

Using the Blog Server Control vs. Using the Individual Blog Server Controls

When you use the Blog server control to display a blog, you only have to deal with one server control and its settings. This makes adding a blog to your site quick and easy.

Using the Individual Blog server controls to display a Blog, allows you the flexibility to display the parts of a Blog you want to display. For example, you might to display the Blog entries and a Blog roll, but not display a Blog calendar. In addition, using the individual Blog server controls allows you to further define the display details of each control.

Blog Server Control

The Blog server control allows you to quickly add a blog to a Web form. It has all of the items commonly found on a blog page in one server control. These items include blog posts, a comments link, a blog roll, blog categories, recent blog

posts, the RSS feed for a blog, an archive and a blog calendar. If you want to learn more about each item, see the User Manual section on "Blogs"

Using the Blog server control allows you to easily maintain the overall look of the blog. When you change display properties, such as the background color or the font, they change for the whole blog not just a piece of it.

While you can change display properties, you can not change the location of each item in the server control. The title always appears on the top, with the tagline below it. The blog posts always appear to the left and the calendar, blog roll, blog categories, recent blog posts, RSS feed, archive and blog roll to the right.


Ektron Medical Blog

Blogging your health!

[Add Post](#)


What is Guillain-Barre Syndrome (GBS)?

(General Information, Neurology)

 [edit Permanent link](#)

Guillain-Barré (Ghee-yan Bah-ray) Syndrome, also called acute inflammatory demyelinating polyneuropathy and Landrys ascending paralysis, is an inflammatory disorder of the peripheral nerves - those outside the brain and spinal cord. It is characterized by the rapid onset of weakness and, often, paralysis of the legs, arms, breathing muscles and face. GBS is the most common cause of rapidly acquired paralysis in the United States today, affecting one to two people in every 100,000.

The disorder came to public attention briefly when it struck a number of people who received the 1976 Swine Flu vaccine. It continues to claim thousands of new victims each year, striking any person, at any age, regardless of gender or ethnic background.



[Subscribe](#)

<< March 2006 >>						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Blogroll

[Bill's Blog](#)

Archive

[October 2006](#)
[March 2006](#)

Categories

[General Information](#)
[Cardiology](#)

Recent Posts

[New Test Post](#)
[Welcome to ektron Medical](#)

The Blog server control is easy to use. Simply add it a Web form and set the BlogPath property to the folder location in the CMS400.NET. Next, choose whether to enable RSS and whether to show the header and tagline.

If you want to further customize the look of the blog, you can edit the Blog.css file located in `webroot\siteroot\Workarea\csslib`.


Blog Server Control Properties

The following table explains the properties of the Blog control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type				
ArchiveMode	<p>Select whether the archive appears in <i>month</i> format or <i>year</i> format. The default is month.</p> <table><thead><tr><th>Month View</th><th>Year View</th></tr></thead><tbody><tr><td><div>Archive</div><div>October 2006</div><div>March 2006</div></td><td><div>Archive</div><div>2006</div></td></tr></tbody></table>	Month View	Year View	<div>Archive</div> <div>October 2006</div> <div>March 2006</div>	<div>Archive</div> <div>2006</div>	String
Month View	Year View					
<div>Archive</div> <div>October 2006</div> <div>March 2006</div>	<div>Archive</div> <div>2006</div>					
Authenticated	<p>Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also</i>: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p>	String				
BlogID	<p>The ID of the blog in CMS400.NET. For example: 41.</p> <p>If you don't know the ID of the blog, use the CMS Explorer to browse to it. See <i>Also</i>: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p>	Integer				
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See <i>Also</i>: "Caching with Server Controls" on page 42.</p>	Integer				
DateToStart	<p>The date of the last blog entries you want to appear. For example, if you want to display blog entries for January 1, 2007 and before, you enter 1/1/2007. Clicking the dropdown box provides you with a calendar.</p>	DateTime				

Property	Value	Data Type
DynamicParameter	Gets or sets the Querystring parameter to read a Blog ID dynamically. Set to "None - Use Default" if you want to always display the default blog. None - Use Default - use the default Blog ID list. ID - reads a Blog's ID dynamically. ekfrm - reads a form block's ID dynamically.	String
EditorHeight	Sets the height of the blog editor in pixels.	Integer
EditorWidth	Sets the width of the blog editor in pixels.	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
JavascriptEditorHTMLMode	Set to True to allow users to edit their posts and replies in HTML. The default is False . See Also: "Allowing HTML Editing in the Javascript Editor" on page 151	String
Language	Set a language for viewing blog posts. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
MaxResults	Set the maximum number of posts to display. If set to 0 (zero), there is no limit. If set to -1, all posts for the day are shown. If set to -2, all posts for the month are shown, If set to -3, the control uses the # of Post Visible setting in the Workarea. The default is -3 . For information on the Workarea setting # of post visible , see the User Manual section "Working With Folders and Content" > "Blogs" > "Blog Properties" > "Blog Folder Properties" > "# Posts Visible"	Integer

Property	Value	Data Type
PostParameter	<p>Works like the <code>DynamicParameter</code> for content blocks. When <code>id</code> is selected, this server control passes the blog post ID as a URL parameter.</p> <p>The default setting is id.</p> <p>Blank - the list of blog posts is static. The links in the blog posts are inactive.</p> <p>id - the id of the blog post is passed to the URL as a parameter.</p> <p>None use default - the list of blog posts is static. The links in the blog posts are inactive.</p>	String
RecentPosts	The number of post links contained in the Recent Posts list. The default is 5 (five).	Integer
ShowHeader	<p>Shows the title and tagline when set to True.</p> <p>The default is True.</p> <p>True - show title and tagline.</p> <p>False - do not show header and tagline.</p>	Boolean
ShowRSS	<p>Displays the icon for the RSS feed () when set to True. The default is True.</p> <p>True - show RSS feed icon.</p> <p>False - do not show RSS feed icon.</p>	Boolean
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is Span.</p> <p>Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <code><div></code> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

BlogEntries Server Control

The BlogEntries server control allows you to display a list of multiple blog posts on a Web form. By using this server control, you can add a blog posts to a Web form without having to add items normally contained in a blog.

Using this server control in combination with the other individual blog server controls, allows you the flexibility to change the appearance of this control without changing other blog controls. In addition, you can layout the controls any way you like, unlike the Blog server control. Below is an example of a BlogEntries server control.



You can use the BlogCalendar and BlogEntries server controls together to allow blog readers to choose which day's blog posts they see.

BlogEntries Server Control Properties

The following table explains the properties of the BlogEntries control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
BlogPostParameter	Works like the <code>DynamicParameter</code> for content blocks. When <code>id</code> is selected, this server control passes the blog post ID as a URL parameter. If you do not set this parameter to <code>id</code> , you will not be forwarded to the blog post's page when you click on any links in the post. The default setting is <code>id</code> . Blank - the list of blog posts is static. The links in the blog posts are inactive. id - the id of the blog post is passed to the URL as a parameter. None use default - the list of blog posts is static. The links in the blog posts are inactive.	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42 .	Integer
DateToStart	The date of the last blog entries you want to appear. For example, if you want to display blog entries for January 1, 2006 and before, you enter 1/1/2006. Clicking the dropdown box provides you with a calendar.	DateTime
DisplayXslt	Specify an external XSLT file. See Also: "The OverrideXslt and DisplayXslt Properties" on page 113 .	String
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean

Property	Value	Data Type
Language	Set a language for viewing the blog entries. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
MaxResults	Set the maximum number of posts to display. If set to 0 (zero), there is no limit. If set to -1, all posts for the day are shown. If set to -2, the all post for the month are shown, If set to -3, the control uses the # of Post Visible setting in the Workarea. The default is -3 . For information on the Workarea setting # of post visible , see the User Manual section "Working With Folders and Content" > "Blogs" > "Blog Properties" > "Blog Folder Properties" > "# Posts Visible"	Integer
ShowHeader	Shows the title and tagline when set to True. The default is True . True - show title and tagline. False - do not show header and tagline.	Boolean
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

BlogPost Server Control

The BlogPost server control allows you to display an individual blog post on a page.

There are two ways this server control displays a blog post. First, if a user is logged in as a CMS400.NET user or a Membership user, the control displays the blog post, comments and comments form on the Web page. The second way is if a user is not logged in, the control displays only the blog post.

NOTE the Enable Comments property in the workarea must be enabled for comments and the comments form to appear.

BlogPost Server Control Properties

The following table explains the properties of the BlogPost control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
DefaultContentID	The ID of a default blog post that appears where you inserted this server control if no other content block is identified, or is not available. If you don't know the ID number of the blog post, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DisplayXslt	Specify an external XSLT file. See Also: "The OverrideXslt and DisplayXslt Properties" on page 113.	String
DynamicParameter	To make this blog post dynamic, select id . When you do, this server control uses the blog post passed as a URL parameter.	String
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the blog posts. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
ShowType	Shows a blog post's content and its comments or just the blog post's comments. The default is Content . Content - Show a blog post's content and its comments. Description - Show a blog post's comments only.	String

Property	Value	Data Type
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

BlogCalendar Server Control

The BlogCalendar server control allows a user to display a calendar on a Web page and associate it with a blog. When a blog calendar is associated with a blog, the days that have blog posts are highlighted on the calendar.

You can use a BlogCalendar server control with a BlogEntries server control. This gives site visitors a way to view blog posts for a given day.

BlogCalendar Server Control Properties

The following table explains the properties of the BlogCalendar control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String

Property	Value	Data Type
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the calendar. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

BlogRoll Server Control

The BlogRoll server control displays the blog roll for a blog on a Web form. A blog roll is a list of links to other blogs. Blog rolls are created in the Workarea for each blog. This control reads that list and displays a clickable list of links. For more information on the blog roll, see the User Manual Section "Blogs" > "Blog Properties" > "The Blog Roll".

Typically this control is displayed along side other individual Blog server controls. Using the individual server controls, such as the BlogRoll server control, allows you more flexibility for layout and appearance than using the Blog server control.

BlogRoll Server Control Properties

The following table explains the properties of the BlogRoll control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also</i> : "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See <i>Also</i> : "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See <i>Also</i> : "Caching with Server Controls" on page 42 .	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the Blog roll. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

BlogCategories Server Control

The BlogCategories server control displays a blog's categories on a Web form. Blog categories are created in the Workarea for each blog. This control reads that

list and displays a clickable list of links. When a link is clicked it displays all the posts associated with the category.

Typically this control is displayed along side other individual Blog server controls. Using the individual server controls, such as the BlogCategories server control, allows you more flexibility for layout and appearance than using the Blog server control.

BlogCategories Server Control Properties

The following table explains the properties of the BlogCategories control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See <i>Also: "Caching with Server Controls" on page 42.</i>	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the Blog categories. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer

Property	Value	Data Type
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

BlogRecentPosts Server Control

The BlogRecentPosts server control displays a list of recent blog posts links on a Web form. when a link is clicked it takes you to the blog post. You can control how many recent post links are displayed by setting the `NumberOfPosts` property.

Typically this control is displayed along side other individual Blog server controls. Using the individual server controls, such as the BlogREcentPosts server control, allows you more flexibility for layout and appearance than using the Blog server control.

BlogRecentPosts Server Control Properties


The following table explains the properties of the BlogRecentPosts control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String

Property	Value	Data Type
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the blog's Recent Posts list. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
NumberOfPosts	Sets the number of post links to display. The default is 5 (five).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

BlogRSS Server Control

The BlogRSS server control displays the icon () for the blog's RSS feed on the Web form. When the icon is clicked, the RSS feed for the blog appears.

Typically this control is displayed along side other individual Blog server controls. Using the individual server controls, such as the BlogRSS server control, allows you more flexibility for layout and appearance than using the Blog server control.

BlogRSS Server Control Properties

The following table explains the properties of the BlogRSS control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also</i> : "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See <i>Also</i> : "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See <i>Also</i> : "Caching with Server Controls" on page 42 .	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the RSS feed. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

BlogArchive Server Control

The BlogArchive server control displays a list of past months or years that have posts. This server control can be used with the BlogEntries server control and the calendar control to allow site users to scan for older Blog posts.

Ektron Medical Blog

Blogging your health!

What is Guillain-Barre Syndrome (GBS)?

(General Information, Neurology)

Guillain-Barré (Ghee-yan Bah-ray) Syndrome, also called acute inflammatory demyelinating polyneuropathy and Landrys ascending paralysis, is an inflammatory disorder of the peripheral nerves - those outside the brain and spinal cord. It is characterized by the rapid onset of weakness and, often, paralysis of the legs, arms, breathing muscles and face. GBS is the most common cause of rapidly acquired paralysis in the United States today, affecting one to two people in

<<

March 2006

>>

Sun	Mon	Tue	Wed	Thu	Fri	Sat
			<u>1</u>	2	3	4
5	6	<u>7</u>	8	9	10	11
12	<u>13</u>	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Archive

[October 2006](#)
[March 2006](#)

By clicking a month in the archive, the calendar changes and the entries appear for that month.

BlogArchive Server Control Properties

The following table explains the properties of the BlogArchive control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
ArchiveMode	<p>Select whether the archive appears in <i>month</i> format or <i>year</i> format. The default is month.</p> <div><div><div>Month View</div><div><div>Archive</div><div><div><div><div><div></div></div></div><div><div><div>October 2006</div><div>March 2006</div></div></div></div></div></div><div><div>Year View</div><div><div>Archive</div><div><div><div><div></div></div></div><div><div><div>2006</div></div></div></div></div></div></div></div>	String

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
Hide	Used to hide the control in design time and run time. True = Hide the control False = Show the control	Boolean
Language	Set a language for viewing the Blog's archive. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

BreadCrumb Server Control

There are two kinds of breadcrumbs: sitemap breadcrumbs and history breadcrumbs.



Sitemap breadcrumbs show you the path through the sitemap to the current page, like the path shown at the top of the page of a Yahoo! directory listing. Sitemap breadcrumbs always show the same path for the same page, regardless of how that page is reached. To use a sitemap breadcrumb, see "[FolderBreadcrumb Server Control](#)" on page 82.



History breadcrumbs more accurately fulfill the breadcrumb trail metaphor. They display an edited version of the browser's history - links to each of the most recently visited pages in the order that they were visited.

The BreadCrumb server control displays history breadcrumbs. Your site visitors can use the breadcrumb trail to jump to previously visited pages.

The breadcrumb trail can appear vertically or horizontally on a Web page. In addition, you can define how many breadcrumbs are left for site visitors to navigate back. You can also use an image and Alt text instead of, or in addition to, text to display the breadcrumb trail.

NOTE If a site visitor revisits a page that is in the breadcrumb trail already, the breadcrumb trail reverts back to that point in the trail. For example, if you visit the following pages Dev > FAQ > WebForm > PR, and then return to FAQ, the breadcrumb trail looks like this: **Dev > FAQ**.

Breadcrumb Trail Type	Example
Horizontal; text only	Dev > WebForm > pr > FAQ > Meta*
Horizontal with images	
Horizontal with images and Alt text	

Breadcrumb Trail Type	Example
Vertical; text only	<pre> > Dev > WebForm > pr > FAQ > Meta* </pre>
Vertical with images	 <p>A vertical breadcrumb trail where each item is enclosed in a blue rounded rectangle. The items are 'DEV', 'PR', 'FAQ', 'Search', and 'Meta', each followed by a right-pointing chevron '>'.</p>
Vertical with images and Alt text	 <p>A vertical breadcrumb trail where each item is enclosed in a blue rounded rectangle. The items are 'DEV', 'PR', 'FAQ', 'Search', and 'Meta', each followed by a right-pointing chevron '>'. A mouse cursor is hovering over the 'Meta' button, which has a small tooltip box labeled 'Meta' appearing below it.</p>

You should add a BreadCrumb server control to every Web form in your site. Doing this ensures that a breadcrumb trail is created for each Web page.

This subsection contains the following topics:

- ["BreadCrumb Properties Apply to Web Forms, Not Web Pages" on page 74](#)
- ["Types of BreadCrumb Server Control Properties" on page 74](#)
- ["BreadCrumb Server Control Properties" on page 75](#)
- ["Using the BreadCrumb Server Control" on page 79](#)
- ["Making a Content Block's Title Appear in the Breadcrumb Trail" on page 80](#)

BreadCrumb Properties Apply to Web Forms, Not Web Pages

Keep in mind that your Web site is made up of Web forms. Each page depends on a form to determine much of its appearance. Forms and pages have a parent > child relationship. That is, any form can be associated with any number of pages.

When defining breadcrumb properties, you define them for a *form*. All pages that use that form inherit its breadcrumb properties.

Assume, for example, that a Web form is used for the Human Resources section of your Web site. You might use the title **Human Resources** to identify the form in the BreadCrumb server control. Whenever a page is visited that uses that form, **Human Resources** appears on the breadcrumb trail -- that is, the form title appears, not the individual page.

Types of BreadCrumb Server Control Properties

BreadCrumb server control properties are divided into two categories:

- ["Properties that Determine a Breadcrumb Trail's Appearance" on page 74](#)
- ["Properties that Determine How Pages Associated with the Form Appear on a Breadcrumb Trail" on page 75](#)

Properties that Determine a Breadcrumb Trail's Appearance

For each Web form, you can customize the breadcrumb trail. For example, you might want the breadcrumb trail to appear horizontally on one Web form and vertically on another.

Property	Determines this aspect of the breadcrumb trail's appearance on this form	For more information, see
CurrentPageIndicator	Symbols or characters to identify the current page	"CurrentPageIndicator" on page 76
DisplayStyle	Whether it appears horizontally or vertically	"DisplayStyle" on page 76
LinkLastItem	Whether the last item is a hyperlink	"LinkLastItem" on page 77
LinkTarget	The type of window that appears when a user clicks an item	"LinkTarget" on page 77
MaxItems	The maximum number of items	"MaxItems" on page 78

Property	Determines this aspect of the breadcrumb trail's appearance on this form	For more information, see
Mode	whether the breadcrumb trail appears as hyperlinked text or plain text.	"Mode" on page 78
Separator	Symbols or characters used to separate items	"Separator" on page 78

Properties that Determine How Pages Associated with the Form Appear on a Breadcrumb Trail

Use these properties define how any page that uses this form appears within a breadcrumb trail. It does not matter where the trail appears.


Note that you can use text, an image, or both to describe the form within the breadcrumb trail. If you use both, the image appears first, followed by the text.

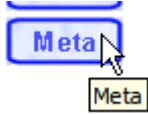
Property	Determines this aspect of Web form when it appears on breadcrumb trail	For more information, see
DisplayTitle	Text to describe it	"DisplayTitle" on page 76
IconAlt	"Alt" text associated with image specified in IconPath property	"IconAlt" on page 77
IconPath	Image to define it	"IconPath" on page 77

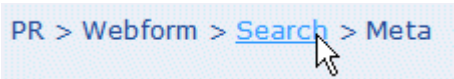
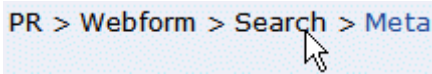

BreadCrumb Server Control Properties

The BreadCrumb server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.


Property	Response	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CurrentPageIndicator	<p>If desired, enter one or more symbols or characters that identify the current page in the breadcrumb trail. See example below.</p> <p style="text-align: center;">Dev > WebForm > pr > FAQ > Meta </p> <p>These characters appear <i>after</i> the image or title that identifies the current page in the breadcrumb trail.</p>	String
DefaultContentID	Gets or Sets the ContentID for the BreadCrumb display title.	Integer
DisplayStyle	Indicate how to display the breadcrumb trail: horizontally or vertically. The default is Horizontal. See Also: "Breadcrumb Trail Type" on page 72	String
DisplayTitle	<p>Enter text to describe this Web form when it appears in the breadcrumb trail. For example, if the Web form whose properties you are defining is used for all Human Resources pages on your site, enter Human Resources. See Also: "BreadCrumb Properties Apply to Web Forms, Not Web Pages" on page 74</p> <p>The default is Title.</p> <p>If you define an image in the IconPath property below, the image appears in the trail, followed by this text.</p>	String
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.	String
Hide	<p>Used to hide the breadcrumb trail in design time and run time.</p> <p>True = Hide breadcrumb trail</p> <p>False = Show breadcrumb trail</p>	Boolean

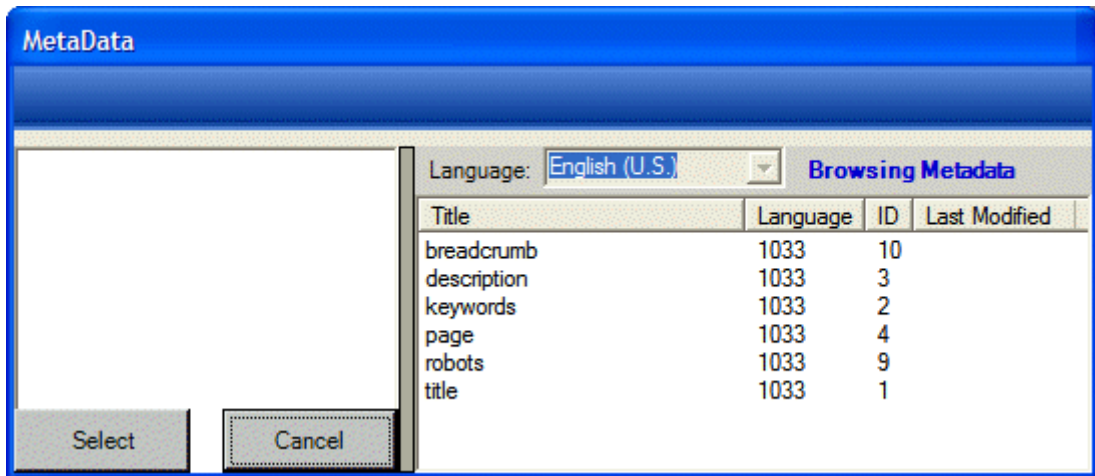
Property	Response	Data Type
IconAlt	<p>If you define an image in the IconPath property, enter any "Alt" text that should appear when a site visitor hovers the cursor over that image. Here is an example.</p> 	String
IconPath	<p>If you want the breadcrumb trail to display an image to identify this Web form, enter the path to the image.</p> <hr/> <p>Important: The image location must be relative to the Web root.</p> <hr/> <p>For example: \cms400Min\Workarea\Images\bc_meta_icon.gif</p> <p>On the breadcrumb trail, the image precedes any text defined in the DisplayTitle property.</p>	String
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkLastItem	<p>Use this property to determine whether the last breadcrumb item appears as a hyperlink on this Web form. If this property is set to true, and a user clicks the item, the current page reappears.</p> <p>True = last item is a hyperlink False = last item is an image and/or text only; the user cannot click on it</p>	Boolean
LinkTarget	<p>Determines the type of window that appears on this Web form when a user clicks an item in the breadcrumb trail. The default is _self.</p> <p>_Self - opens in same window _Top - opens in parent window _Blank - opens in new window _Parent - opens in the parent frame</p>	String

Property	Response	Data Type
MaxItems	<p>Enter the maximum number of items in the breadcrumb trail on this Web form.</p> <p>The default is 5.</p> <p>If you set a value of 1 or greater and the user navigates beyond that number of pages, only the most recent pages appear. The older pages disappear from the trail.</p> <p>To allows an unlimited number of breadcrumb trail items on this Web form, enter zero (0).</p>	Integer
MetadataName	<p>Specify the name of a Metadata Type that you want to associate with the page. For more information, see "Breadcrumb Metadata Type" on page 79</p>	String
Mode	<p>Allows you to make the breadcrumb trail appear as non-hyperlinked plain text. The default is Normal.</p> <p>Normal = breadcrumb trail is hyperlinked</p>  <p>DisplayOnly = breadcrumb trail is plain text</p> 	String
Separator	<p>Enter one or more characters to separate the items in a breadcrumb trail on this Web form.</p> <p>The default character is the greater than sign (>).</p> 	String
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

BreadCrumb Metadata Type

To associate a Metadata Type with the BreadCrumb server control, follow these steps.

1. In the properties window for the BreadCrumb server control, click on the **MetadataName** property.
2. Click the Ellipses button ().
If you are not logged in, log in now using the CMS Explorer window.
3. The Browsing Metadata screen appears.



4. Select a Metadata Type to apply to the BreadCrumb server control.
5. Set the DynamicParameter to ID. This allows the metadata to reflect what ever content block appears on the page.

Using the BreadCrumb Server Control

Add the BreadCrumb server control to each Web form for which you want to create a breadcrumb trail. Follow these steps to use the BreadCrumb server control.

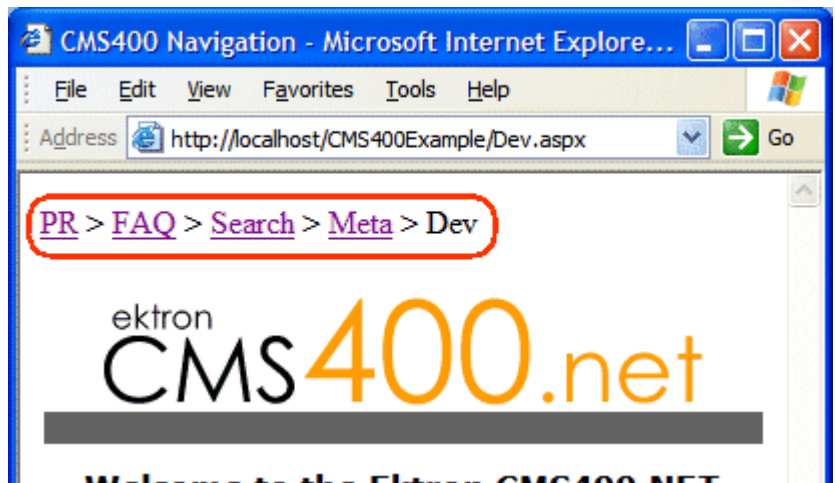
WARNING!

It is recommended that you do not add a BreadCrumb server control to a Web form that contains a Calendar server control. When both server controls appear on the same page and a site visitor clicks on different months, the BreadCrumb trail could look like this: Calendar > Calendar > Calendar > Calendar > Calendar. This happens because each time a site visitor clicks on a month, you are in effect opening a new Web form.

1. Open a Web form for which you want to create a breadcrumb trail.
2. Drag and drop the BreadCrumb server control onto an appropriate location of the Web form.
3. Set the **DisplayTitle** property. See ["DisplayTitle" on page 76](#).

NOTE To customize the look of your breadcrumb trail, configure the other properties. For more information, see "BreadCrumb Server Control Properties" on page 75.

4. Save the Web form.
5. Repeat steps one through four for all Web forms to be included in the breadcrumb trail.
6. Open a browser.
7. View a Web page with a BreadCrumb server control in it.
8. Navigate to the rest of the Web pages that contain BreadCrumb server controls.
9. As you move around, the pages are added to the breadcrumb trail.



Making a Content Block's Title Appear in the Breadcrumb Trail

By adding a line or two of code, the breadcrumb trail can display the title of a content block. You can add a line of code to each Web form or if your content is dynamic, add the code once and each time a new piece of content is called dynamically a new breadcrumb is created.

Only one content block per Web form can be used with the breadcrumb. If you have multiple content blocks on a page, select the one that describes the overall page best.

Steps to Use a Content Block's Title as a Breadcrumb

1. Drag and drop a **ContentBlock** server control on a Web form.
2. Choose a **DefaultContentID** for the content block.
3. Drag and drop a **BreadCrumb** server control on the same form.
4. Remove the word **Title** from the **DisplayTitle** property.
5. Add the following line code to the page load event code behind:

```
BreadCrumb1.DisplayTitle = ContentBlock1.EkItem.Title
```

6. Build the project.
7. View the Web form in a Web browser.

The following is an example content block titles used as breadcrumbs:



The word Title appears in the breadcrumb trail when the DisplayTitle property is left empty. Even with the added code, if the breadcrumb loads before the content block, the breadcrumb has no information in the DisplayTitle property and therefore displays the word title by default.

If the word Title appears in the breadcrumb trail, check to make sure the word Title does not appear in the DisplayTitle property. Next, if Title still appears, add the following line of code to the page load event of the code behind:

```
ContentBlock1.Fill()
```

The code in the page load event should now look like this:

```
ContentBlock1.Fill()  
Breadcrumb1.DisplayTitle = ContentBlock1.EkItem.Title
```

This ensures that the content block information is loaded first.

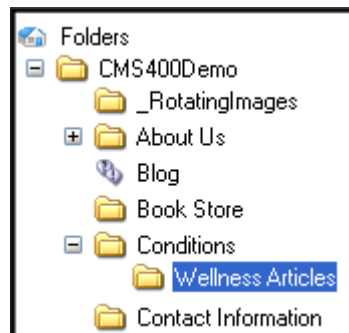
FolderBreadcrumb Server Control

As mentioned in the BreadCrumb server control section, there are two kinds of breadcrumbs: sitemap breadcrumbs and history breadcrumbs. Ektron handles history breadcrumbs with the BreadCrumb server control. See *Also*: "[BreadCrumb Server Control](#)" on page 72.

Sitemap breadcrumbs show you the path through the sitemap to the current page, like the path shown at the top of the page of a Yahoo! directory listing. Sitemap breadcrumbs always show the same path for the same page, regardless of how that page is reached.

The FolderBreadcrumb server control displays sitemap breadcrumbs. This type of trail is typically made up of the folder path for the content. For example, the FolderBreadcrumb below matches the folder structure where the content is located.

Home >> Conditions >> Wellness Articles



It is important to note that the FolderBreadcrumb server control does not simply read your folder structure and display its path. It reads breadcrumb information set in the Workarea's folder properties for each folder.

Administrators define a folder's breadcrumbs on the Breadcrumb tab located in the Workarea folder properties. See *Also*: The Administrator Manual > "Managing Content" > "Managing Content Folders" > "Breadcrumbs"

This subsection contains the following topics:

- "[BreadCrumb Server Control Properties](#)" on page 82
- "[Using the BreadCrumb Server Control](#)" on page 84

BreadCrumb Server Control Properties

The BreadCrumb server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	String
DefaultContentID	Display the breadcrumb trail for the folder in which the given content ID resides. To use this property, breadcrumb information for a folder must be defined in the Workarea > [Specified Folder] > Folder properties > Breadcrumb tab.	Integer
DefaultFolderID	The folder ID for which you want the breadcrumb trail to display. If a DefaultContentID is given, it overrides this property. To use this property, breadcrumb information for a folder must be defined in the Workarea > [Specified Folder] > Folder properties > Breadcrumb tab.	Integer
DisplayStyle	Indicate how to display the breadcrumb trail: horizontally or vertically. The default is Horizontal.	String
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.	String
Hide	Used to hide the breadcrumb trail in design time and run time. True = Hide breadcrumb trail False = Show breadcrumb trail	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	Determines the type of window that appears on this Web form when a user clicks an item in the breadcrumb trail. The default is _self . _Self - opens in same window _Top - opens in parent window _Blank - opens in new window _Parent - opens in the parent frame	String

Property	Description	Data Type
Mode	Allows you to make the breadcrumb trail appear as non-hyperlinked plain text. The default is Normal . Normal = breadcrumb trail is hyperlinked DisplayOnly = breadcrumb trail is plain text	String
Separator	Enter one or more characters to separate the items in a breadcrumb trail on this Web form. The default character is the greater than sign (>).	String
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

Using the BreadCrumb Server Control

Add the FolderBreadcrumb server control to each Web form for which you want to create a breadcrumb sitemap. Follow these steps to use the FolderBreadcrumb server control.

1. In the CMS400.NET Workarea, define breadcrumb information for a folder. This done on a folder's Breadcrumb tab in the folder properties. See the Administrator Manual > "Managing Content" > "Managing Content Folders" > "Breadcrumbs" > "Adding a Breadcrumb" for the steps to add a breadcrumb.
2. Open a Web form for which you want to create a folder breadcrumb map.
3. Drag and drop the FolderBreadcrumb server control onto an appropriate location of the Web form.
4. Set the **DefaultContentID** or **DefaultFolderID** property. If using DefaultContentID, make sure the content is in a folder where breadcrumb information has been defined. If using DefaultFolderID, make sure breadcrumb information has been defined for that folder.
5. Save the Web form.
6. Open a browser.
7. View a Web page with the FolderBreadCrumb server control in it.

8. The FolderBreadcrumb now appears on your site as a sitemap breadcrumb.



BusinessRules Server Control

The BusinessRules server control allows you to add a Business Rule created in the Workarea to a Web form. It also allows you to add a place holder for a Business Rule. Then, a CMS400.NET administrator can create a business rule at a later date.

For an explanation of Business Rules and how they work, see the Administrator Manual > "Business Rules".

BusinessRules Server Control Properties

The following table explains the properties of the Business Rules server control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.	String
Hide	Used to hide a control in design time and run time. True = Hide Business Rule False = Show Business Rule	Boolean
Language	Set a language for viewing form content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
RuleID	The ID of the Ruleset to evaluate. If you don't know the ID number of the Ruleset, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

Property	Value	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String






























































































Calendar Server Control

The Calendar server control displays a calendar that was created in the Workarea. By using the Calendar server control's properties, you can select multiple views of the calendar to display. See Also: ["DisplayType" on page 91](#)

You can adjust a calendar's look and feel by using a style sheet. See Also: ["Calendar Style Sheet" on page 93](#) and ["calendarStyles.css" on page 912](#).

The Calendar server control can output the raw XML data of the calendar. You can use this XML data to create a completely customized calendar.

Below is a Calendar displayed on the Web Page. When logged in, you can add and view calendar events. When logged out, calendar events only appear.

Login Status	Calendar display																																										
Logged in	<div><div>vember</div><div>December 2004</div><div>Januai</div><table><tr><th>Sun</th><th>Mon</th><th>Tue</th><th>Wed</th><th>Thu</th><th>Fri</th><th>Sat</th></tr><tr><td></td><td></td><td></td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr><tr><td></td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr><tr><td></td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr><tr><td></td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td></td></tr></table></div>	Sun	Mon	Tue	Wed	Thu	Fri	Sat				1 	2 	3 	4 		6 	7 	8 	9 	10 	11 		13 	14 	15 	16 	17 	18 		20 	21 	22 	23 	24 	25 		27 	28 	29 	30 	31 	
Sun	Mon	Tue	Wed	Thu	Fri	Sat																																					
			1 	2 	3 	4 																																					
	6 	7 	8 	9 	10 	11 																																					
	13 	14 	15 	16 	17 	18 																																					
	20 	21 	22 	23 	24 	25 																																					
	27 	28 	29 	30 	31 																																						

Login Status	Calendar display																																										
Logged Out	<div><div>November</div><div>December 2004</div><div>January</div><table><tr><th>Sun</th><th>Mon</th><th>Tue</th><th>Wed</th><th>Thu</th><th>Fri</th><th>Sat</th></tr><tr><td></td><td></td><td></td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td></tr><tr><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td></td></tr><tr><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td></td></tr><tr><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td></td><td></td></tr></table></div>	Sun	Mon	Tue	Wed	Thu	Fri	Sat				1	2	3	4		6	7	8	9	10	11	13	14	15	16	17	18		20	21	22	23	24	25		27	28	29	30	31		
Sun	Mon	Tue	Wed	Thu	Fri	Sat																																					
			1	2	3	4																																					
	6	7	8	9	10	11																																					
13	14	15	16	17	18																																						
20	21	22	23	24	25																																						
27	28	29	30	31																																							

A calendar can be *static* or *dynamic*.

- A static calendar is selected in the server control properties.
- A dynamic calendar is passed to the page by a URL parameter.

This subsection contains the following topics:

- ["Calendar Server Control Properties" on page 89](#)
- ["Calendar Style Sheet" on page 93](#)

Calendar Server Control Properties


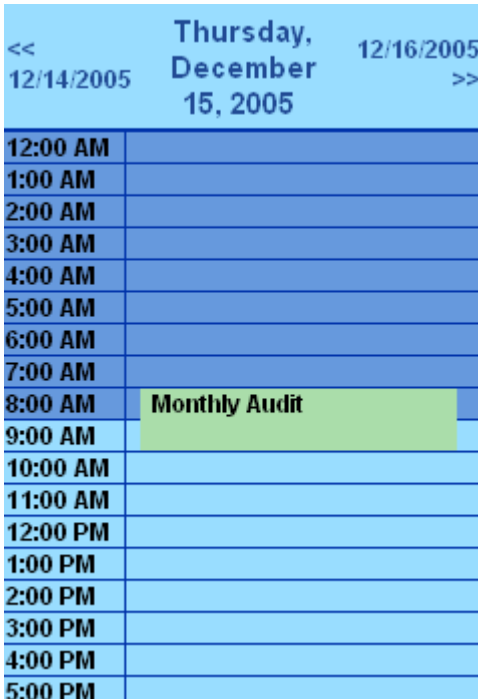
The Calendar server control properties are described in this table.

NOTE

There are additional settings to help define the calendar in the Ektron CMS400.NET Workarea. These include display weekends, forward only, show all label, etc. For information on these properties, see Ektron CMS400.NET Administrator's Manual > Calendars > Managing Calendar Modules > Viewing Calendar Modules.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	String
DefaultCalendarID	The ID number of the calendar that appears where you inserted this server control. If you don't know the ID number of the calendar, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer

Property	Value	Data Type
DisplayType	<p>Specify the Calendar view type. The default is Month.</p> <ul style="list-style-type: none"> Month - A 30 day calendar month view.  <ul style="list-style-type: none"> Day - Single day view. 	String

Property	Value	Data Type
DisplayType (Con't)	<ul style="list-style-type: none"> Inline - Display events that occur from the start date of the calendar through the end date. <div> <div>Thursday, December 01, 2005 - Saturday, December 31, 2005</div> <div>12/15/2005</div> <div> Monthly Audit Business Office Start: 8:00 AM End: 10:30 AM This is to test a recurring calendar event. </div> <div>12/21/2005</div> <div> Christmas Party Event Room Start: 7:30 PM </div> </div> MonthUpWithEvent - Displays a month calendar. Events appear to the right of the calendar when you click on a day. <div> <div> <div><< December, Jan >></div> <div>Nov 2005 >></div> <div> <div>Mon Tue Wed Thu Fri</div> <div> <div></div><div></div><div></div><div>1</div><div>2</div> </div> <div> <div>5</div><div>6</div><div>7</div><div>8</div><div>9</div> </div> <div> <div>12</div><div>13</div><div>14</div><div>15</div><div>16</div> </div> <div> <div>19</div><div>20</div><div>21</div><div>22</div><div>23</div> </div> <div> <div>26</div><div>27</div><div>28</div><div>29</div><div>30</div> </div> <div>2004 .. 2005 .. 2006</div> </div> <div> <div>12/21/2005</div> <div>Christmas Party</div> <div>Location: Event Room</div> <div>Start: 7:30 PM</div> <div>End: 11:00 PM</div> <div>Come to the Christmas p.</div> </div> </div> </div> <p>Note: The URL parameters <code>hevtype</code> and <code>evhighlight</code> do not work. They are reserved for future use.</p>	
DynamicParameter	To make this calendar dynamic, select calendar_id . When you do, this server control uses the calendar passed as a URL parameter.	String
EndDate	Specify the end date the calendar displays. Used in conjunction with the <code>StartDate</code> property. For example, you have a calendar that shows only classes for the next three month semester.	Date

Property	Value	Data Type
Hide	Used to hide output of calendar in design time and run time. True = Hide calendar False = Display calendar	Boolean
Language	Set a language for viewing calendar. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
ShowXML	At runtime, output the raw XML data instead of the calendar. With this XML data, you can apply your own XSL to create your own custom display formats. Default is False . True = Show raw XML data False = Show Calendar	Boolean
StartDate	Specify the start date the calendar displays. Used in conjunction with the EndDate property. For example, you have a calendar that shows only classes for the next three month semester.	Date
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an inline portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

Calendar Style Sheet

Before you insert a Calendar into your Ektron CMS400.NET template, Ektron recommends referencing the style sheet `\webroot\siteroot\Workarea\csslib\calendarStyles.css` in your template. Copy the style sheet to your webroot folder then add it to your project. Next create a reference to `calendarStyles.css`. For example:

```
html>
<head>
```

```
<title>Company Calendar</title>
  <link href="calendarStyles.css" rel="stylesheet" type="text/
css" />
</head>
```

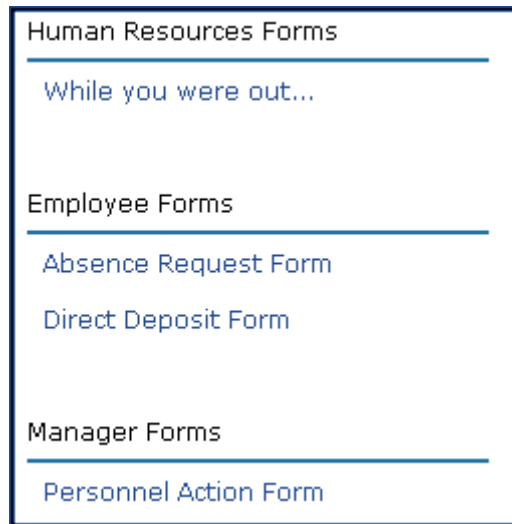
This can also be accomplished via drag and drop. If you drag and drop the `calendarStyles.css` from the Solution Explorer to your Web form, the reference is created in the HTML automatically.

For a description of the classes contained in `calendarStyles.css`, see ["calendarStyles.css" on page 912](#).

Collection Server Control

The Collection server control displays a collection, that is a list of content links that is created in the Ektron CMS400.NET Workarea. The control lets you customize the display of the collection on a Web page.

When added to a template and visited, collections can look like this (The following illustration shows three collections on the same page of a sample site.) You can modify the display by editing its server control properties.



This section contains the following topics:

- ["Collection Server Control Properties" on page 95](#)
- ["Example of ecmNavigation Display" on page 101](#)
- ["Example of ecmTeaser Display" on page 102](#)
- ["Retrieving the XML Structure of a Collection" on page 103](#)
- ["Using the Collection Server Control Programmatically Example" on page 104](#)

Collection Server Control Properties


The Collection server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
AddText	Override the derived controls default Add-menu button text. For example, "Add HTML Content" could be changed to "Add Report."	String
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached.</p> <p>For example, to cache data for five minutes, set the <code>CacheInterval</code> property to 300. See Also: "Caching with Server Controls" on page 42.</p> <hr/> <p>Important! If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Integer
ContentParameter	Checks the <code>QueryString</code> for this value and replaces the collection with a content block when specified. Leave blank to always display the Collection.	String
DefaultCollectionID	<p>The ID of a collection that appears where you insert this server control if no other collection is identified, or is not available.</p> <p>If you don't know the ID number of the collection, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p>	Integer

Property	Description	Data Type
DisplayXslt	<p>Determines how the information on the page is displayed.</p> <p>None-databind only</p> <p>ecmNavigation - lists the title of every content block in the collection</p> <p>See Also: "Example of ecmNavigation Display" on page 101</p> <p>ecmTeaser - lists the title of every content block in the collection plus the content summary</p> <p>See Also: "Example of ecmTeaser Display" on page 102</p> <p>Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p>Important! If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String
DynamicParameter	<p>To make this collection dynamic, select coll_id. When you do, this server control uses the collection passed as a URL parameter.</p>	String

Property	Description	Data Type
EnablePaging	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of collection items while controlling the amount of screen space. To accomplish this, the collection display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you set this property to true, and the number of collection items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="549 598 958 826" data-label="Image"> </div> <p>So, for example, if a collection has 9 items and the <code>MaxResults</code> property is set to 4, the screen displays only the first four items. When the site visitor clicks [Next], he sees items 5, 6, 7 and 8, etc.</p> <hr/> <p>Important! If the <code>EnablePaging</code> property is set to True, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Boolean
GetHtml	Set to True if you want to retrieve and display content (html body) for all content blocks in the collection. For example, to display content inside a web server control such as a GridView.	Boolean
Hide	<p>Used to hide output of collection in design time and run time.</p> <p>True = Hide collection</p> <p>False = Display collection</p>	Boolean

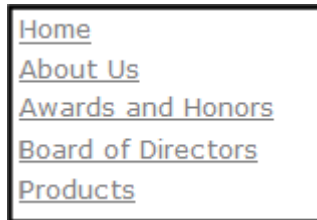
Property	Description	Data Type
IncludeIcons	<p>Choose whether to display icons next to the collection list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Image con] variable is used in an EkML file.</p> <hr/> 	Boolean
Language	Set a language for viewing the collection. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	<p>Defines the way a link acts when a link is clicked. Choices are:</p> <p>_blank - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p>_self - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned BASE target.</p> <p>_parent - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p>_top - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>	String

Property	Description	Data Type
MarkupLanguage	<p>Identify the template markup file that controls the display of the collection. For example, mycollectionmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \siteroot\workarea\customfiles\markup\mycollectionmarkup.ekml.</p> <p>See Also: "Ektron Markup Language" on page 754 and "collection.ekml" on page 778</p> <hr/> <p>Note: If you enter a valid EkML file, the Displayxslt property value is ignored.</p> <hr/>	String
MaxResults	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, set the EnablePaging property to true.</p> <p>If you do and more than the number of MaxResults are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="548 1137 958 1365" data-label="Image"> </div>	Integer
Random	Set to True if you want to randomly display one collection item. The item changes each time a site visitor views the page.	Boolean
SuppressWrapperTags	This property is set to <code>false</code> because Ajax uses <code><div></code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code> .	Boolean

Property	Description	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is Span.</p> <p>Span - The tag is used to designate an inline portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

Example of ecmNavigation Display

The following is an example of a collection being used as a navigation menu.



ecmNavigation XSL code

Below is the XSL code that is used to create ecmNavigation Display. You can use this code as the basis to design your own XSLT.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <table border="0" cellspacing="0" cellpadding="0" width="100%">
    <xsl:for-each select="Collection/Content">
      <tr>
        <td>
          <a>
            <xsl:attribute name="href">
              <xsl:value-of select="QuickLink"/>
            </xsl:attribute>
            <xsl:value-of select="Title"/>
          </a>
        </td>
      </tr>
    </xsl:for-each>
  </table>
</xsl:template>

</xsl:stylesheet>

```

Example of ecmTeaser Display

The following is an example of a collection using the ecmTeaser display style.

Home	Products	Support	Latest News	Careers	Search	Calendar
<h3>Employment Opportunities</h3> <p>Plastic Molder #123 RC International is looking for an experienced plastics molder.</p> <p>Servo Control Engineer #124 RC International is looking for a highly skilled servo control engineer to join our team.</p>						

ecmTeaser XSL code

Below is the XSL code that is used to create ecmTeaser Display. You can use this code as the basis to design your own XSLT.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <table border="0" cellspacing="0" cellpadding="0" width="100%">
    <xsl:for-each select="Collection/Content">
      <tr>
        <td>
          <a>
            <xsl:attribute name="href">
              <xsl:value-of select="QuickLink"/>
            </xsl:attribute>
            <xsl:value-of select="Title"/>
          </a>&#160;

// xml.Append("<xsl:value-of select='substring(DisplayDateModified, 0, string-
length(DisplayDateModified)-11)'" />
        </td>
      </tr>
      <tr>
        <td>
          <xsl:value-of select="Teaser" />
        </td>
      </tr>
      <tr>
        <td>&#160;</td>
      </tr>
    </xsl:for-each>
  </table>

</xsl:template></xsl:stylesheet>

```

Retrieving the XML Structure of a Collection

Retrieving the XML structure of XML content allows for greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a Collection server control onto it.
3. Set the DefaultCollectionID property.
4. Drag and drop a Textbox on the Web form.
5. Set the TextMode property to MultiLine.

NOTE It is also recommended that you set the width of the text box to at least 400px.

6. On the code behind page, add the following line.

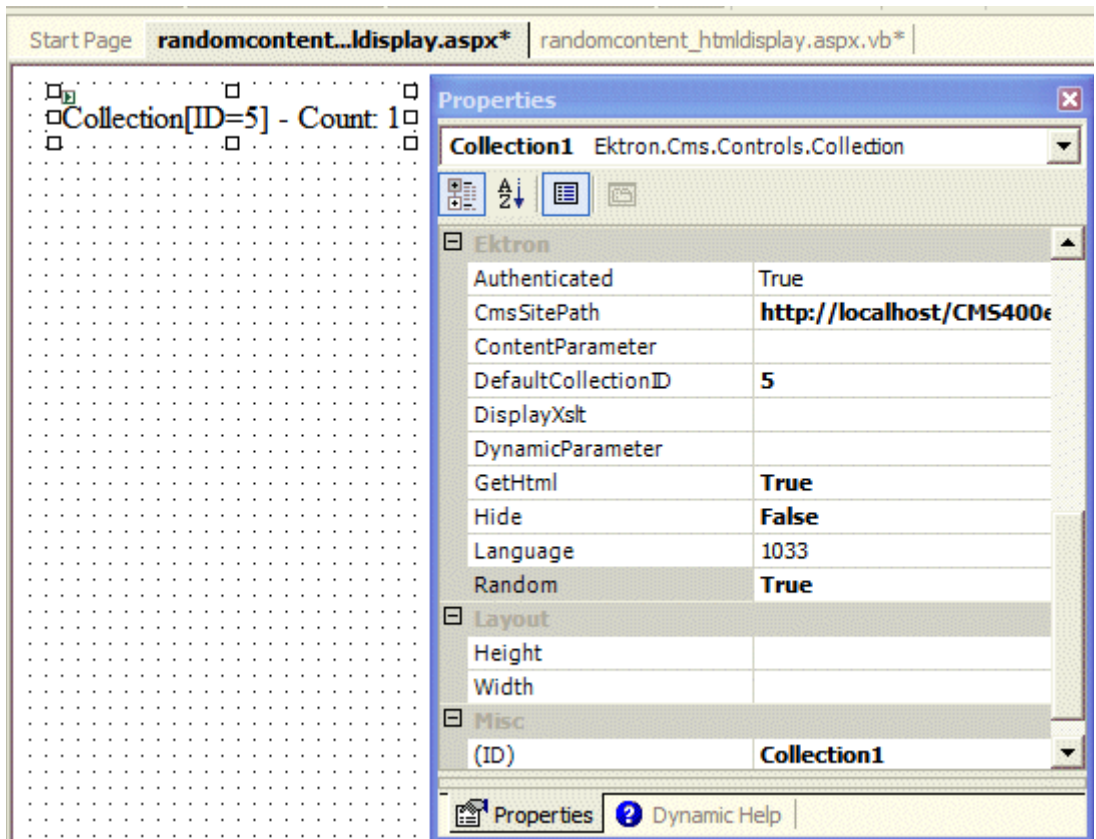
```
Textbox1.Text = Collection1.XmlDoc.InnerXml
```
7. Build the project.
8. View the Web form in a browser.
9. The XML structure of the collection appears in the textbox.

Using the Collection Server Control Programmatically Example

The following is an example of using code behind, and a drag and drop Collection server control to display random content from a collection.

1. Drag and Drop a Collection server control on your Web form.
2. Set the properties in the properties window.

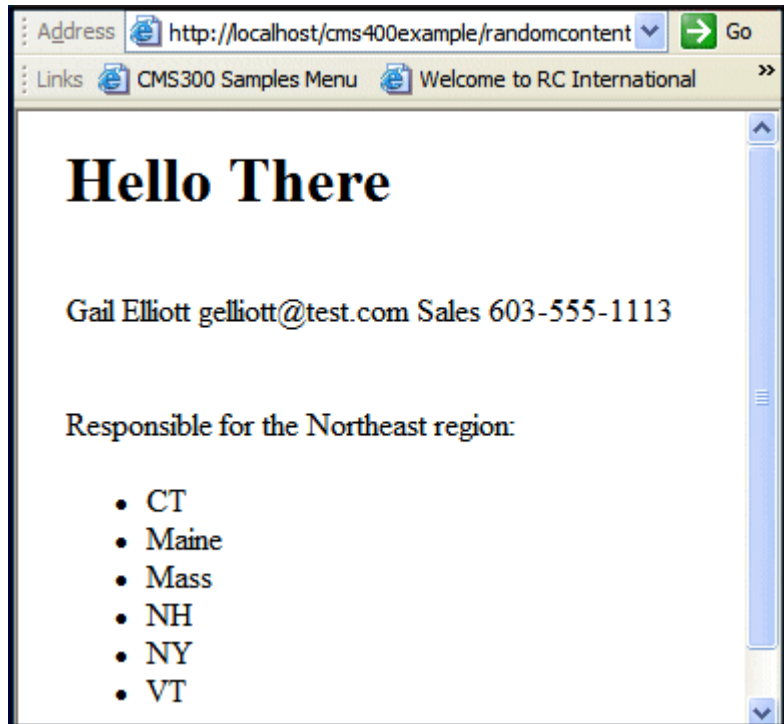
NOTE In this example, the Random property and the GetHtml property must be set to **True**.



3. Add the following code to the code behind.

```
Dim str As String
str = "<h1>Hello There</h1><br>"
str &= Collection1.EkItems(0).Html
Collection1.Text = str
```

4. Build and browse your web form.



Content Analytics Server Control

Use the AnalyticsTracker server control to track statistics about visits to your Web site, such as:

- how often your content was viewed
- how many site visitors viewed for first time, and how many returned
- the most popular referral pages (that is, the Web page from which visitors clicked to arrive on your site)

For more information about Content Analytics, see the **Ektron CMS400.NET** Administrator manual chapter "Content Analytics Feature."

Content Analytics Server Control Properties

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Ektron Properties

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Boolean
DefaultContentID	If desired, enter the ID of a content item being tracked by this server control. If you do, all visits to this content item are tracked for content analytic statistics. If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer
DynamicParameter	To make this content analytic control dynamic, select id . When you do, this server control is attached to the content block passed as a URL parameter.	String

Property	Value	Data Type
Hide	Select False to display this server control on the page. Select True to suppress it.	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Lets you use custom tag.	String
EnableAnalytics	Determines if the Analytics feature is enabled for this page and (if identified) content item. ConfigSpecified - Take the value from the web.config file's <code>enableAnalytics</code> property. True - Enable Analytic tracking False - Disable Analytic tracking See Also: Ektron CMS400.NET Administrator Manual Section "Content Analytics Feature" > "Enabling the Content Analytics Feature."	

ContentBlock Server Control

The ContentBlock server control displays a content block on an Ektron CMS400.NET Web page.

Ektron CMS400.NET has two types of content blocks:

- Static - displays one specified content block
- Dynamic - displays the content block of the ID passed through a URL parameter

In addition, a content block can contain XHTML or XML content.

For an example of using the ContentBlock server control programmatically, see ["Using the ContentBlock Server Control Programmatically Example" on page 115](#)

This subsection contains the following topics:

- ["Static Content Block" on page 108](#)
- ["ContentBlock Server Control Properties" on page 108](#)
- ["Dynamic Content Block" on page 110](#)
- ["XML Content Block" on page 112](#)
- ["The OverrideXslt and DisplayXslt Properties" on page 113](#)
- ["Using the ContentBlock Server Control Programmatically Example" on page 115](#)

Static Content Block

A static content block displays one identified content block on a Web page. The following table explains how to complete the server control properties to insert a static content block on a Web page.

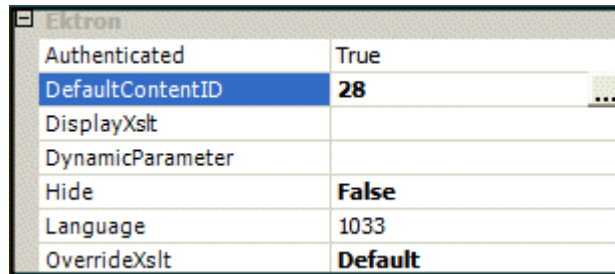
ContentBlock Server Control Properties

The following table explains the properties of the ContentBlock server control.

NOTE	The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.
-------------	--

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
DefaultContentID	<p>The ID of a content block that appears where you insert this server control.</p> <p>If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p> <hr/> <p>Note: If you identify a content block that displays an Office document which will be published as HTML, make sure the template sets <code></code> tags to display the content as a block.</p> <hr/>	Integer
DisplayXslt	Ignore for a non-XML content block.	String
DynamicParameter	Select None - Use Default . This parameter is used for dynamic content blocks.	String
Hide	<p>Used to hide a content block in design time and run time.</p> <p>True = Hide content block</p> <p>False = Show content block</p>	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean

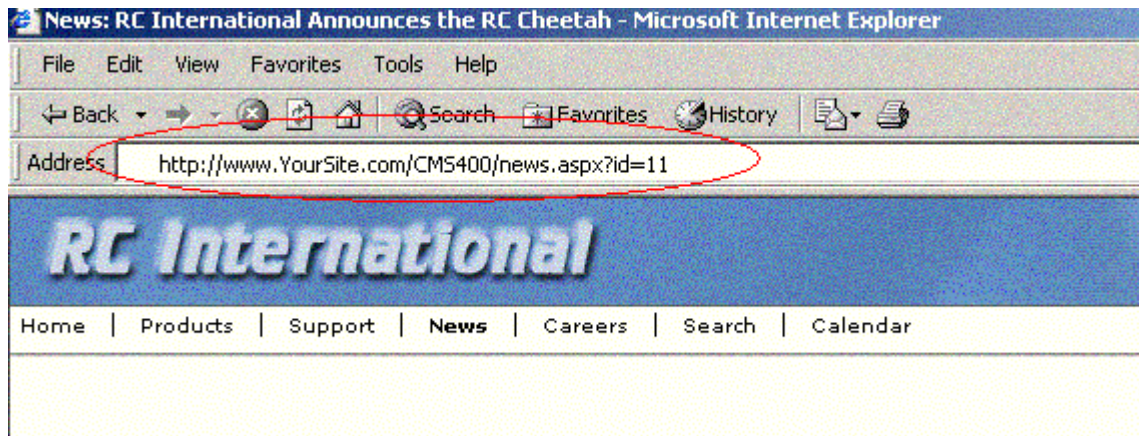
Property	Value	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String



In the above example, the server control retrieves content block id=28 and displays it in the browser.

Dynamic Content Block

Use a dynamic content block to display a content block whose ID is passed through a URL parameter. You would use this server control with a dynamic template.



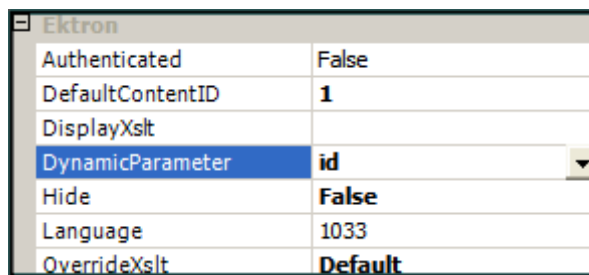
The following table explains how to complete the ContentBlock server control's properties to insert a dynamic content block on a template.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
DefaultContentID	The ID of a content block that appears where you inserted this server control if no other content block is identified, or is not available. If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DisplayXslt	Ignore for a non-XML content block.	String
DynamicParameter	Select id . When you do, this server control uses the content block passed as a URL parameter.	String
Hide	Used to hide a content block in design time and run time. True = Hide content block False = Show content block	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean

Property	Value	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

The following example shows how the ContentBlock server control can be used to pass the id as a URL parameter.



When a user clicks a link that passes the content block ID as a URL parameter, that content block appears. If that content block is not available, content block 1 appears.

XML Content Block

Use an XML content block to display an XML content block on an Ektron CMS400.NET Web page.

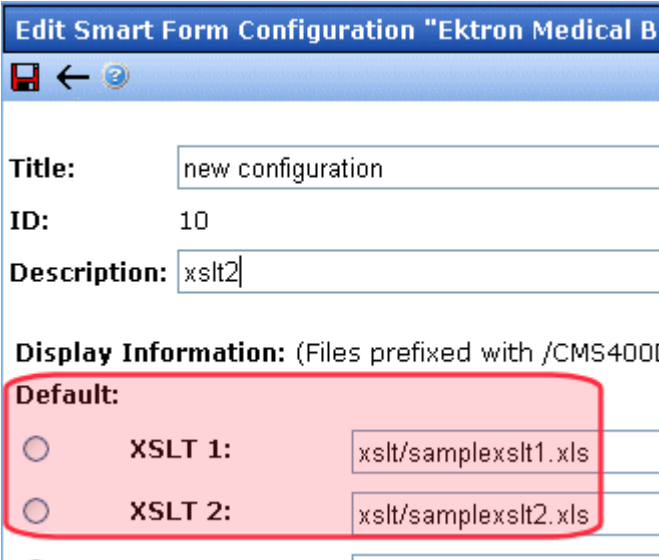
NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see [VisualStudio help](#).

Property	Value	Data Type
Authenticated	<p>Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p>	String

Property	Value	Data Type
DefaultContentID	The ID of a content block that appears where you inserted this server control if no other content block is identified, or is not available. If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DisplayXslt	Specify an external XSLT file. See Also: "The OverrideXslt and DisplayXslt Properties" on page 113	String
DynamicParameter	To make this content block dynamic, select id . When you do, this server control uses the content block passed as a URL parameter.	String
Hide	Used to hide a content block in design time and run time. True = Hide content block False = Show content block	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

The OverrideXslt and DisplayXslt Properties

The DisplayXslt property is optional. If used, it specifies an external XSLT file. If the DisplayXslt property is not defined, the OverrideXslt property specifies an XSLT identified in the Edit Smart Form Configuration screen. The following table provides more information about these properties.

To use this display XSLT	Enter this value for XSLT	Enter this value for Override XSLT
<p>The default XSLT specified in the Edit Smart Form Configuration screen (illustrated below). Note that, in this example, the XSLT Packaged option is the default XSLT, since it is selected.</p> <p>XSLT Packaged is the XSLT from the Edit Smart Form Configuration screen (that is, the XSLT created in the Data Designer).</p> 		Default
XSLT 1 from the Edit Smart Form Configuration screen		1
XSLT 2 from the Edit Smart Form Configuration screen		2
XSLT 3 from the Edit Smart Form Configuration screen		3

To use this display XSLT	Enter this value for XSLT	Enter this value for Override XSLT
An absolute or relative path to an XSLT file.	An external XSLT file not specified in the Edit Smart Form Configuration screen. For example <code>sample.xslt</code>	does not matter - if XSLT value exists, OverrideXSLT is ignored

See Also: The "Managing XML" chapter of the **Ektron CMS400.NET** Administrator manual

Retrieving the XML Structure of an XML Content Block

Retrieving the XML structure of XML content allows for greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a Content Block server control onto it.
3. Set the DefaultContentID to an XML content block.

WARNING!

This does not work with HTML content blocks, as there is no XML structure to output.

4. Drag and drop a Text box on the Web form.
5. Set the TextMode property to MultiLine.

NOTE

It is also recommended that you set the width of the text box to at least 400px.

6. On the code behind page, add the following line.
`TextBox1.Text = ContentBlock1.EkItem.Html`
7. Build the project.
8. View the form in a browser.
9. The XML structure of the content block appears in the text box.

Using the ContentBlock Server Control Programmatically Example

The following code will display a content block:

NOTE Before adding these lines of code, you need to drag and drop a literal box on your Web form.

```
Dim MyContentBlock As New ContentBlock
MyContentBlock.DefaultContentID = 8
MyContentBlock.Page = Page
MyContentBlock.Fill()
Literal1.Text = MyContentBlock.EkItem.Html
```

To display a content block with the content block title, do the following:

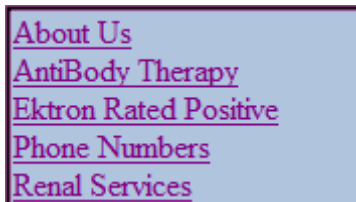
NOTE Before adding these lines of code, you need to drag and drop a two literal boxes on your Web form.

```
Dim MyContentBlock As New ContentBlock
MyContentBlock.DefaultContentID = 8
MyContentBlock.Page = Page
MyContentBlock.Fill()
Literal1.Text = MyContentBlock.EkItem.Title
Literal2.Text = MyContentBlock.EkItem.Html
```

ContentList Server Control

The ContentList server control is used to display a list of content blocks on a Web page. In contrast to a List Summary, where content must be in a specified folder, the ContentList server control uses content from any CMS400.NET folder.

When added to a template and visited, a ContentList Summary looks like this.



Depending on the setting you choose for DisplayXslt, you can change the amount of information displayed for each content block. See Also: ["DisplayXslt" on page 119](#).

There are two options for the ContentList server control to display a content list. First, by defining a content list in the metadata of a content block and using that content block's ID in the **DefaultContentID** property. See Also: ["Using the DefaultContentID and MetaTag Properties" on page 122](#).

Second, by adding a comma delimited list of content blocks to **ContentIds** property. See Also: ["Using the ContentID Property to Display a Content List" on page 126](#).

Using the DefaultContentID and MetaTag process, an administrator typically sets up the Workarea portion of the process. Then a developer adds the server control to a Web form, assigns the ID of the content block and the metadata name to the appropriate properties. Because the content list in this process is assigned to a content's metadata, you can pass the content ID dynamically in a Web form and display a list for each content block you have defined.

Using the ContentIds process, a developer adds ContentList server control to Web form. Then, defines a list of content IDs in the ContentIds property. Once the server control is deleted or the list in the ContentIds property is deleted, It is not available and will have to be created again.

This subsection contains the following topics:

- ["ContentList Server Control Properties" on page 117](#)
- ["Using the DefaultContentID and MetaTag Properties" on page 122](#)
- ["Using the ContentID Property to Display a Content List" on page 126](#)

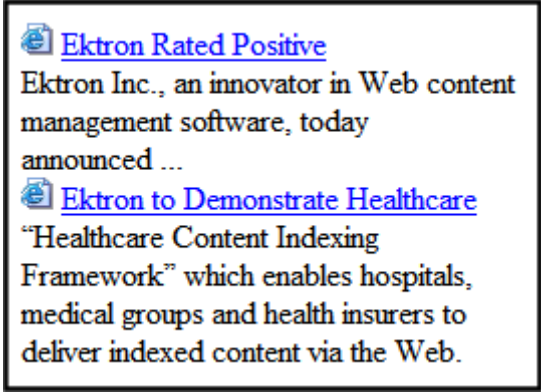
ContentList Server Control Properties

The ContentList server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42	Integer
ContentIds	A comma delimited list of content blocks IDs. See Also: "Using the ContentID Property to Display a Content List" on page 126	String
DefaultContentID	Set content id value. Once set, content IDs are generated from the MetaTag value for this content. See Also: "MetaTag" on page 121 "Using the DefaultContentID and MetaTag Properties" on page 122	Integer
Direction	Determines how to order the hyperlinks on the list. The sort field is determined by the OrderKey property. ascending - they are arranged A, B, C or 1,2,3. descending - they are arranged. Z,Y,X or 3,2,1 If sorting by date, descending puts the most recent first.	String

Property	Description	Data Type
DisplayXslt	<p>Determines how information on the page is displayed.</p> <p>None - databind only</p> <p>ecmNavigation - lists the title of each content block</p> <p>See Also: "Example of ecmNavigation Display" on page 101</p> <p>ecmTeaser - lists the title of each content block plus the content summary</p> <p>See Also: "Example of ecmTeaser Display" on page 102</p> <p>ecmUnOrderedList - sorts the list in no particular order. Shows the title and content summary</p> <p>Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p>Important! If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String
DynamicParameter	<p>Uses the Querystring parameter to read a content ID dynamically.</p> <p>None - Use Default - use the default content ID list.</p> <p>ID - reads a content block's ID dynamically.</p> <p>ekfrm - reads a form block's ID dynamically.</p>	String
GetHtml	<p>Set to True if you want to retrieve and display content (html body) for all content blocks in the list summary. For example, to display content inside a web server control such as a GridView.</p> <p>True = Get and display HTML for each content block in the list summary</p> <p>False = Do not get and display HTML.</p>	Boolean
Hide	<p>Used to hide output of the content list in design time and run time.</p> <p>True = Hide results</p> <p>False = show results</p>	Boolean

Property	Description	Data Type
Includelcons	<p>Choose whether to display icons next to the content list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Imagelcon] variable is used in an EkML file.</p> <hr/>  <p>The screenshot shows two links, each preceded by a small icon. The first link is 'Ektron Rated Positive' followed by the text 'Ektron Inc., an innovator in Web content management software, today announced ...'. The second link is 'Ektron to Demonstrate Healthcare' followed by the text '“Healthcare Content Indexing Framework” which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.'</p>	Boolean
Language	Set a language for viewing the list summary. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p>_blank - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p>_self - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p>_parent - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p>_top - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>	String

Property	Description	Data Type
MarkupLanguage	<p>Identify the template markup file that controls the display of the content list. For example, mycontentlistmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \siteroot\workarea\customfiles\markup\mycontentlistmarkup.ekml.</p> <p>See Also: "Ektron Markup Language" on page 754 and "contentlist.ekml" on page 781</p> <hr/> <p>Note: If you enter a valid EkML file, the <code>Displayxslt</code> property value is ignored.</p> <hr/>	String
MetaTag	<p>Specify a Metadata definition whose type is Content Selector. When you do, the associated list of content items will appear where you place the server control.</p> <hr/> <p>Warning! You cannot insert other metadata types.</p> <hr/> <p>This works with the <code>DefaultContentID</code> property.</p> <p>See Also:</p> <p>"DefaultContentID" on page 118</p> <p>"Using the DefaultContentID and MetaTag Properties" on page 122</p> <p>For more information about using metadata to assign a list of related content to a content item, see the Ektron CMS400.NET Administrator manual section "Managing Content" > "Metadata" > "Types of Metadata" > "Related Content Metadata."</p>	Integer
OrderKey	<p>Sort the list by one of the values.</p> <p>Title - the order of the content's title.</p> <p>DateModified - the last date the content was modified.</p> <p>DateCreated - the date the content was created.</p> <p>LastEditorFname - the last editor's first name.</p> <p>LastEditorLname - the last editor's last name.</p> <p>OrderOfThelds - preserves the content ID order based on the list in the <code>ContentIds</code> property.</p>	String (must be one of the values)

Property	Description	Data Type
Random	Set to True if you want to randomly display one content block in the specified folder. The content changes each time a user views the page. True - randomly display one content block in the specified folder. False - display the content list normally.	Boolean
SuppressWrapperTags	This property is set to <code>false</code> because Ajax uses <code><div></code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code> .	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an inline portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

Using the DefaultContentID and MetaTag Properties

You define a content list in the metadata of a content block. Then, using the ContentList server control, you can display the list of content blocks associated with the content. The ContentList server control pulls information defined in the metadata of the content to display the list of content.

The steps below show an example of using the DefaultContentID and MetaTag properties to create a content list.

1. In the CMS400.NET Workarea, create a metadata definition, in the **Name** field, enter a title of the data. This name is used in the MetaTag property for the server control.
2. Fill in all of the standard fields. See the **Ektron CMS400.NET** Administrator Manual section "Managing Content" > "Metadata" > "Working with Metadata Definitions" > "Adding a Metadata Definition" > "Metadata Definition Fields".

The **Type** must be **Content Selector**.

Add Metadata definition

Name: TestContentList

Type: Searchable Property

Editable: Searchable Property

Display in Ektron Explorer: Searchable Property, Meta Tag, HTML Tag, Collection Selector, **Content Selector**, File Selector, Hyperlink Selector, Image Selector, ListSummary Selector

3. Add the definition to all appropriate folders. See the **Ektron CMS400.NET** Administrator Manual section "Managing Content" > "Metadata" > "Working with Metadata Definitions" > "Assigning Metadata to a Folder"
4. For each content item to which you want to associate related content, access its **Metadata** tab and identify the related item(s).
For example, a list of content blocks associated with motorcycle helmets. You want this list to appear whenever a helmet is being viewed.
To set this up, you would
 - edit the content block
 - click its **Metadata** tab
 - find the metadata definition for the content list
 - click **Change**

Content **Summary** **Metadata** **Schedule** **Comment** **Web Alerts** **Templates**

TestContentList: (Content)

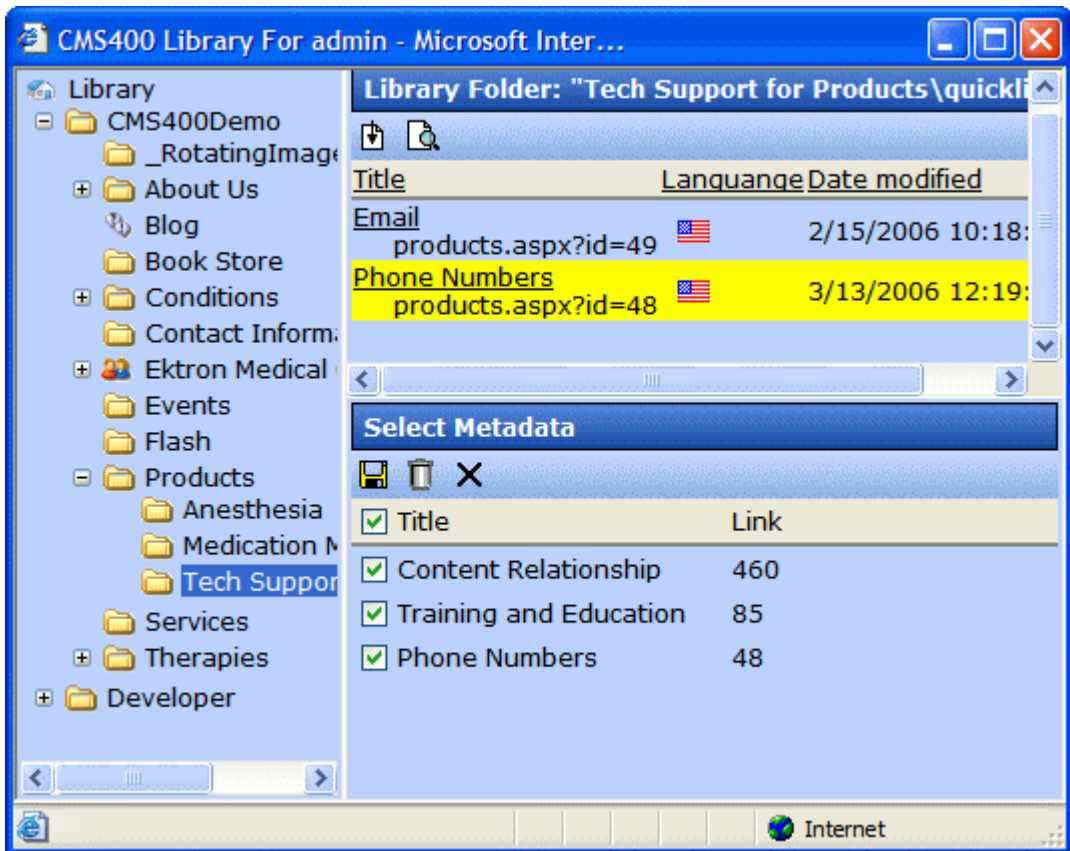
None selected

Change Clear

Search Data

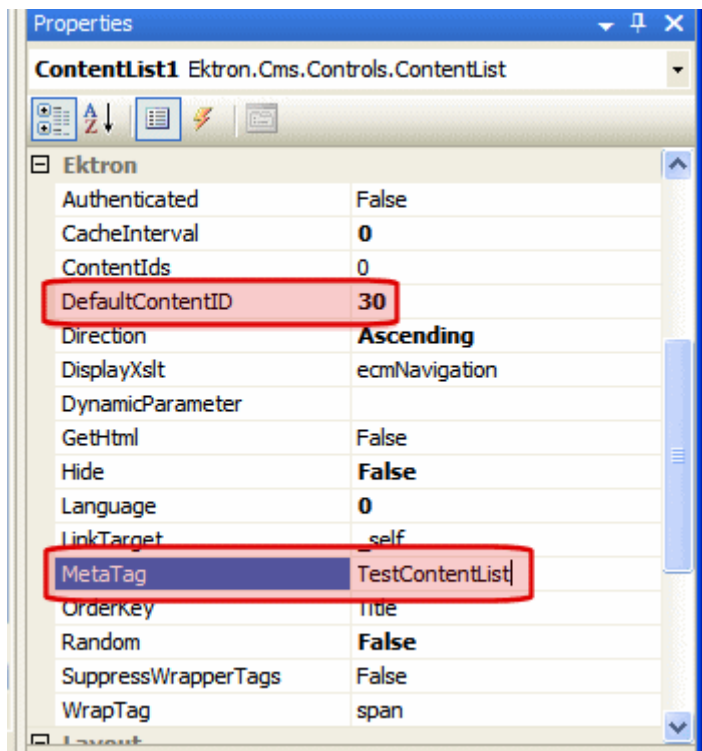
wellness: (No Selection) ▼

- select the content blocks to be in the list

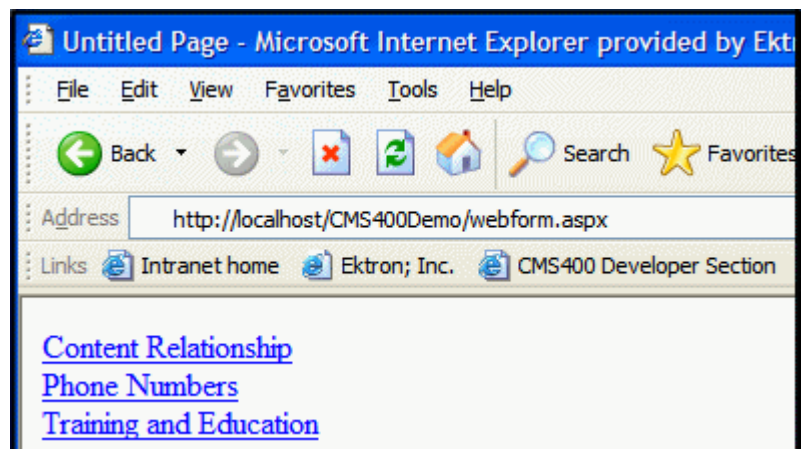


- On a Web form in Visual Studio, add the ContentList server control.
- Set the **DefaultContentID** property to the content block's ID.

7. Add the name of the Metadata field in the Workarea to the **MetaTag** property.

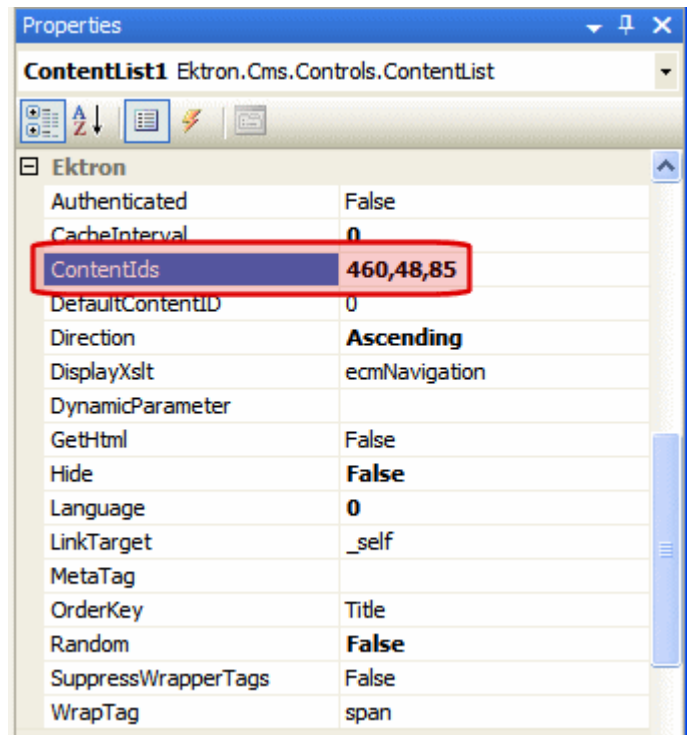


8. Set all other properties using the table in "ContentList Server Control Properties" on page 117.
9. Once the Web form is saved, navigate to the page and the content list is displayed.

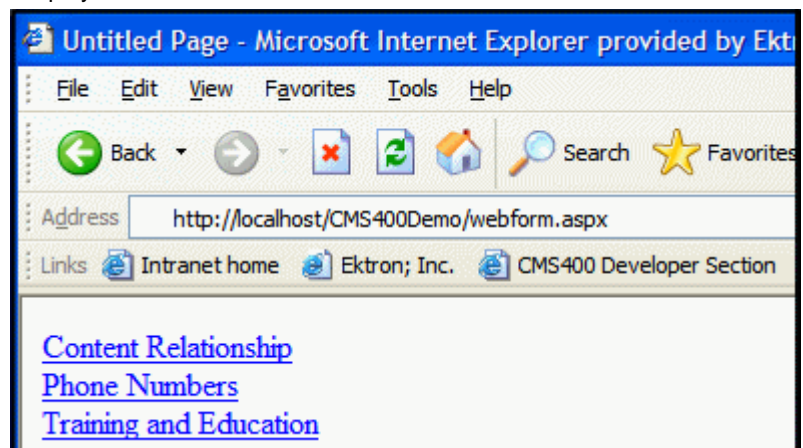


Using the ContentID Property to Display a Content List

1. On a Web form in Visual Studio, add the ContentList server control.
2. Add a comma delimited list of content IDs to the **ContentIds** property.



3. Set all other properties using the table in "ContentList Server Control Properties" on page 117.
4. Once the Web form is saved, navigate to the page and the content list is displayed.



ContentReview Server Control

The ContentReview server control has two ways it can be used. The first, places a star based rating scale on any page of your Web site. The scale lets Ektron CMS400.NET collect feedback by giving site visitors the opportunity to rate a particular content item. An item is rated when a visitor clicks on a star. Depending on the XSLT chosen, visitors can also submit review comments.

The second way the ContentReview control can be used is to retrieve and display visitor reviews and comments. In this mode, the control can display reviews of a content item or reviews by a specific site or membership user. See Also: ["Displaying Content Reviews On the Site" on page 131](#) and ["Displaying a User's Reviews On the Site" on page 132](#).

Ektron CMS400.NET maintains and reports content reviews and ratings data. For more information, see the **Ektron CMS400.NET** Administrator Manual section "Managing Content" > "Content Rating."

ContentReview Server Control Properties

The following table explains the properties of the ContentReview server control.

NOTE [The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.](#)

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Boolean
DefaultContentID	The ID of a content block being rated by this server control. If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer

Property	Value	Data Type
DisplayXslt	<p>Select the type of review to display, or enter the path to a custom XSLT. For additional information, see "Using the DisplayXSLT Property" on page 129</p> <p>Choices are:</p> <p>Ajax 5 Stars - a five star rating system utilizing Ajax for display.</p> <p>Ajax 5 Stars Comment - a five star rating system utilizing Ajax for display. In addition, when you hover over the stars, a review box pops-up below the stars. Comments that are added are submitted via an Ajax call.</p> <p>Ajax 5 Stars with Increments - a five star rating system with half star increments that utilizes Ajax for display.</p> <p>5 Stars - a five star rating system that allows visitors to add text reviews of content.</p> <p>5 Stars with Increments - a five star rating system with half star increments that allows visitors to add text reviews of content.</p> <p>Review List - displays a list of reviews for content or a user. See Also: "Displaying Content Reviews On the Site" on page 131 and "Displaying a User's Reviews On the Site" on page 132.</p> <p>Path to Custom Xslt - enter the path to a custom XSLT.</p>	String
DynamicParameter	<p>To make this content review control dynamic, select id. When you do, this server control is attached to the content block passed as a URL parameter. When using this control to retrieve a user's reviews, set this property to UserId to make the user's ID dynamic.</p>	String
GetReviews	<p>Returns a list of reviews for content or a user.</p> <p>None - do not return reviews.</p> <p>Content - returns reviews based on the content ID provided in the DefaultContentID property.</p> <p>User - returns reviews based on the User ID provided in the UserId property.</p>	Boolean
Hide	<p>Select False to display this server control on the page. Select True to suppress it.</p>	Boolean
Language	<p>Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p>	Integer


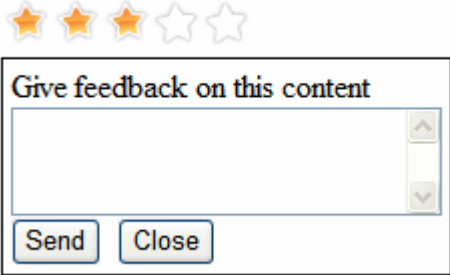

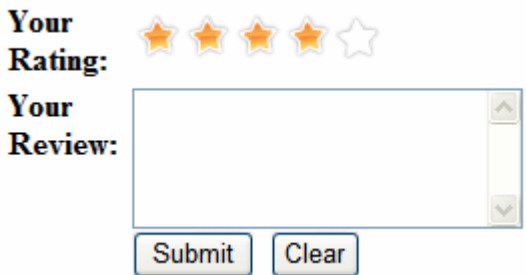
Property	Value	Data Type
MaxReviews	The number of reviews to retrieve if the GetReview property is set to Content or User. 0 (zero) = unlimited results. <i>See Also: "GetReviews" on page 128</i>	Integer
Moderate	Setting this property to True allows CMS400.NET users to moderate reviews. Additional information on moderating reviews can be found in the Ektron CMS400.NET Administrator Manual section "Managing HTML Content" > "Content Rating"	Boolean
RatingsMinimum	Sets a minimum number of reviews and ratings before displaying the average rating. When set to 0 (zero), the average rating is displayed once the content has been rated for the first time.	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
UserId	The user ID for which to get reviews. If left blank, reviews from all users are returned. If a user ID is specified, only reviews for that user are returned. Important: The GetReviews property must be set to User for the control to use this property. <i>See Also: "GetReviews" on page 128</i>	Integer
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Lets you use custom tag.	String




Using the DisplayXSLT Property

The table below shows an example of each XSLT as it appears on a Web page, a description and the XSL file being used. The files are located in [Web Root]\[Site Root]\Workarea\Xslt

IMPORTANT!

If you modify an existing XSL file, Ektron Inc. recommends you change the name of the modified file and enter the path to the new file in the DisplayXSLT Property. The files below are copies of files built, transformed and used, internally.

DisplayXSLT Selection	Description	Example
Ajax 5 Stars	A five star rating system utilizing Ajax for display. File used: rating5star.xsl	
Ajax 5 Stars Comment	A five star rating system utilizing Ajax for display. In addition, when you hover over the stars, a review box pops-up below the stars. Comments that are added are submitted via an Ajax call. File used: rating5starComment.xsl	
Ajax 5 Stars with Increments	A five star rating system with half star increments that utilizes Ajax for display. File used: rating5starinc.xsl	
5 Stars	A five star rating system that allows visitors to add text reviews of content. File used: rating5starAddEdit.xsl	

DisplayXSLT Selection	Description	Example
5 Stars with Increments	A five star rating system with half star increments that allows visitors to add text reviews of content. File used: rating5starincAddEdit.xsl	<p>Your Rating: </p> <p>Your Review: <input type="text"/></p> <p><input type="button" value="Submit"/> <input type="button" value="Clear"/></p>
Review List	Displays a list of reviews for content or a user. File used: ratinglist.xsl See Also: "Displaying Content Reviews On the Site" on page 131 and "Displaying a User's Reviews On the Site" on page 132 .	<p>This example shows content reviews for a content item:</p> <p>Reviews</p> <p>Sample Content Block</p> <p></p> <p>By jmember CMS400.NET provides so many features that I want on my website, all in one integrated package. This is amazing.</p> <p>Sample Content Block</p> <p></p> <p>By AA What a great set of server controls. Our organization is committed to .NET, and this is the most complete .NET CMS by far.</p>

Displaying Content Reviews On the Site

To display content reviews on the site, follow these steps:

1. Drag and drop a ContentReview server control on a Web form.
2. Set the DefaultContentID property to the ID of the content for which reviews will be displayed.

NOTE You can make the reviews for the content item dynamic by leaving the DefaultContentID blank and setting the DynamicParameter to ID.

3. Set the DisplayXSLT property to Review List.

NOTE

If you want to change the look and feel of the review list, modify an existing XSL file or create your own. If you modify an existing XSL, Ektron Inc. recommends saving the modified file with a different name. Then, enter the path to the file in the DisplayXSLT property.

4. Set the GetReviews property to **Content**.
5. Save the Web form.

Below is example of content reviews displayed on a Web page.



Displaying a User's Reviews On the Site

To display reviews for a specific user on the site, follow these steps:

1. Drag and drop a ContentReview server control on a Web form.
2. Set the DisplayXSLT property to Review List.

NOTE


If you want to change the look and feel of the review list, modify an existing XSL file or create your own. If you modify an existing XSL, Ektron Inc. recommends saving the modified file with a different name. Then, enter the path to the file in the DisplayXSLT property.


3. Set the GetReviews property to **User**.
4. Set the UserID property to the ID of the user for which reviews will be displayed.


NOTE You can make the list of reviews for a user dynamic by leaving the UserID blank and setting the DynamicParameter to ID.

5. Save the Web form.

Below is example of a user's reviews displayed on a Web page.

Reviews
Alley Cat Pizzeria

By AA
Not bad, but I like Pizza Hut better.

Beating Cancer With Nutrition

By AA
This book is a good read, but I feel
could have used more information.

Sample Content Block

By AA
What a great set of server controls.
Our organization is committed
to .NET, and this is the most
complete .NET CMS by far.

ContentRating Server Control

WARNING! This server control is deprecated as of the 7.0.1 Release. For optimal performance, Ektron recommends the ContentReview server control. See ["ContentReview Server Control" on page 127](#)

Use the ContentRating server control to place a rating scale on any page of your Web site. The scale lets Ektron CMS400.NET collect feedback by giving site visitors the opportunity to rate a particular content item. A visitor can rank the content on a numerical scale of one through nine, as well as submit a text comment.

The control lets a site visitor rate a content item only once. On subsequent visits to the page, a message informs the visitor that he has previously rated the page, which can optionally display a summary of ratings for it. As a result, the control has two appearance modes: *initial* and *complete*.

Ektron CMS400.NET maintains and reports content ratings data. For more information, see the **Ektron CMS400.NET** Administrator Manual section "Managing Content" > "Content Rating."

Content Rating Server Control Properties

The Ektron-specific Content Rating server control properties are divided into two groups:

- *Ektron* - settings that affect properties, such as the display
- *Ektron labels* - settings that affect the text labels within the control

These properties are described below.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

NOTE

Ektron Properties

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Boolean

Property	Value	Data Type
CompleteGraph Display	<p>Specify a value to indicate how the ratings graph appears if a site visitor has already rated content.</p> <ul style="list-style-type: none"> • horizontal - graph appears to the right of the AlreadyRatedMessage • vertical - graph appears below the AlreadyRated-Message • none - graph does not appear 	String
DefaultContentID	<p>The ID of a content block being rated by this server control. It typically would be the content directly above the rating scale if no other content block is identified, or is not available.</p> <p>If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p>	Integer
DynamicParameter	To make this content rating control dynamic, select id . When you do, this server control is attached to the content block passed as a URL parameter.	String
GraphBarColor	The color of the bars in the graph.	String
Hide	Select False to display this server control on the page. Select True to suppress it.	Boolean
InitialGraphDisplay	<p>Specify a value to indicate how the ratings graph appears if a site visitor has not yet rated content.</p> <ul style="list-style-type: none"> • horizontal - graph appears to the right of the content rating scale • vertical - graph appears below the content rating scale • none - graph does not appear 	String
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer

Property	Value	Data Type
Padding	<p>Enter the number of pixels used to create space between the rating scale and the graph.</p> <hr/> <p>Note: If you do not display the graph, this setting has no effect.</p> <hr/>	Integer
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <code><div></code> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Lets you use custom tag.</p>	String

Ektron Label Properties

The label properties determine the text that appears in the server control. Most of them are illustrated below.

Note that the first illustration describes the initial graph, while the second describes the complete graph.

Initial Graph Display

Feedback Header

How would you rate the usefulness of this content?

Bad Label Poor 1 2 3 4 5 6 7 8 9 Outstanding **Good Label**

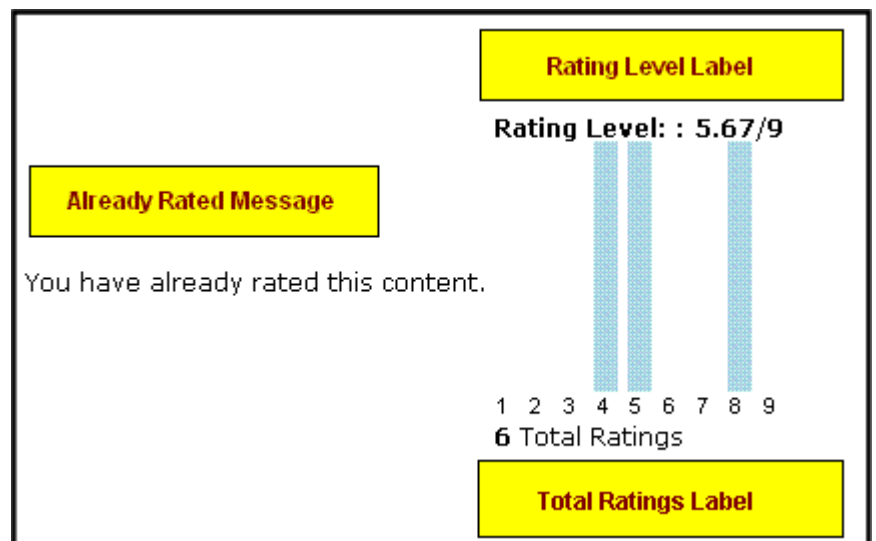
Optional: Tell us why you rated the content this way **User Comments Header**

Characters Remaining Label Characters Remaining: 1024 **Submit**

Please select a rating before submitting your feedback.

Validation Message

Complete Graph Display



Property	Description	Default Text
AlreadyRatedMessage	Appears in place of the rating scale if the user previously rated the page	You have already rated this content.
BadLabel	The low end of the rating scale	Poor
CharsRemainingLabel	Number of additional characters a user may enter into the comment.	Characters Remaining:
CookiesRequired Message	Appears if browser cookies have been disabled	We're sorry, but you must enable cookies in your browser to rate this content.
FeedbackHeader	Introduces rating scale	How would you rate the usefulness of this content?
GoodLabel	The high end of the rating scale	Outstanding
RatingLevelLabel	Appears at the top of the graph	Rating Level:
ThankYouMessage	Appears after site visitor selects a number, optionally enters a comment, then presses Submit .	Thank you for your feedback.
TotalRatingsLabel	Appears below the graph, and lists the number of rating submissions received.	Total Ratings
UserCommentsHeader	Introduces free-text comment area below rating scale	Tell us why you rated the content this way
ValidationMessage	Appears on initial display if user clicks Submit button before selecting rating number	Please select a rating before submitting your feedback.

DesignTimeDiagnostic Server Control

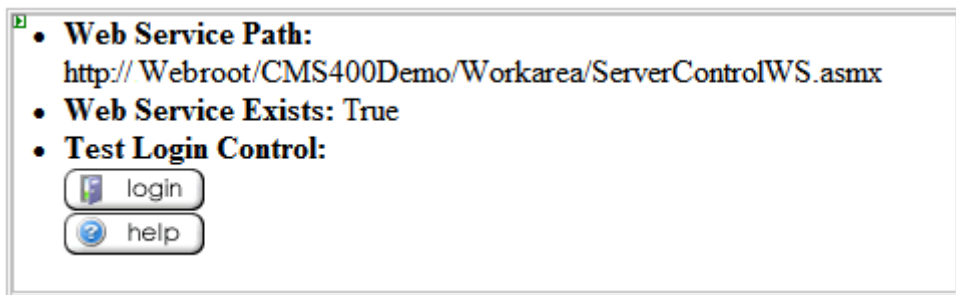
IMPORTANT!

This control is for use in design-time only, nothing is rendered at run-time.

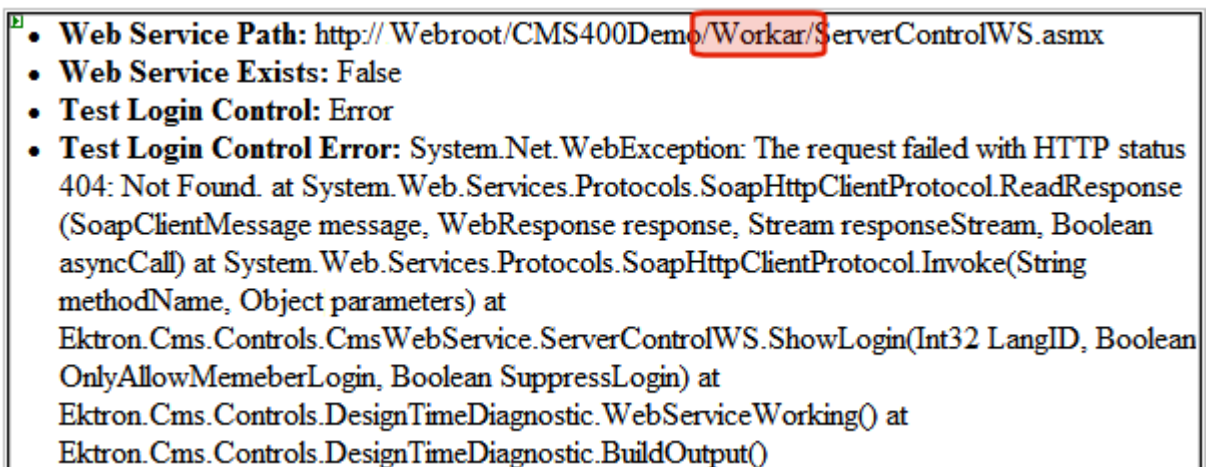
The DesignTimeDiagnostic server control is used to verify the connection to Ektron's server controls Web service. When added to a Web form, this control provides the following information:

- **Web Service Path** - displays the server control Web service path in your web.config file.
- **Web Service Exists** - calls a method in the Web service that returns True when the Web service exists. When it does not, it returns False.
- **Test Login Control** - displays the Login server control to show that the Web service is connected and working properly. When not connecting properly, an error message is displayed.

Below is an example of the control on a page connected to the Web service.



Below is an example of the control on a page not connecting to the Web service. Note that the path is not the correct path. It should be: `http://192.168.0.82/siteroot/Workarea/ServerControlWS.aspx`



DesignTimeDiagnostic Server Control Properties

The DesignTimeDiagnostic server control has no definable ektron properties.

Directory Server Control

The Directory server control lets you customize the behavior of the Taxonomy feature. You place this control on any Web form to display a taxonomy. For more information about the site visitor experience with Taxonomy, see the Ektron CMS400.NET Administrator Manual section "Taxonomy."

Improving Taxonomy Performance

To minimize taxonomy's impact on the performance of your production server, follow these guidelines.

- Use the default page size (50). If you need to customize page size, use Ektron CMS400's [API](#), but do not increase it above 800.
- For large databases, do not increase the value of the `TaxonomyDepth` property above 1. Changing this value dramatically slows down your production server's performance. However, you can increase depth on your staging server for testing purposes.

Directory Server Control Properties

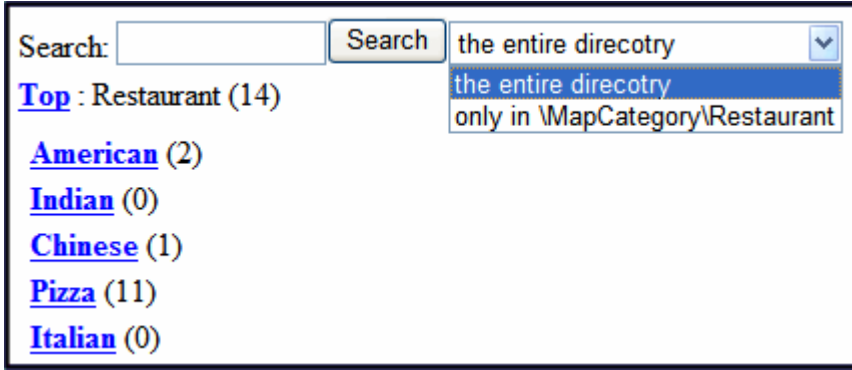
The Directory server control properties are described in this table.

NOTE

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Response	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: " Using CMS Explorer to Browse Your Ektron CMS400.NET Site " on page 15	String
CacheInterval	<p>Sets the amount of time, in seconds, the server control's data is cached. The default is 0 (zero).</p> <p>For example, to cache data for five minutes, set this property to 300. See Also: "Caching with Server Controls" on page 42</p> <hr/> <p>Important! If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Integer

Property	Response	Data Type
DisplayXslt	<p>If you want to use an XSLT to control the display of the taxonomy server control on this page, enter the path to the XSLT file.</p> <hr/> <p>Important! If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.	String
EnableAjax	<p>Set to true to enable Ajax searches. When enabled, the MaxResults property determines the maximum number of results per page.</p> <p>True = Enable Ajax Search (default value)</p> <p>False = Original HTML Search</p>	Boolean
EnablePaging	<p>This property, in conjunction with the MaxResults property, lets site visitors view an unlimited number of taxonomy items while controlling the amount of screen space. To accomplish this, the taxonomy display is limited to the number set in the MaxResults property.</p> <p>If you set this property to true (the default), and the number of taxonomy items exceeds MaxResults, navigation aids appear below the last item. The site visitor uses them to view additional items. See example below.</p> <div data-bbox="601 1104 1011 1334" data-label="Image"> <p>The image shows a rectangular box containing four lines of blue, underlined text: "Bill Lumbergh", "Bob Porter", "Bob Slydell", and "Milton Waddams". Below these names are four navigation links in a lighter blue font: "[First]", "[Previous]", "[Next]", and "[Last]".</p> </div> <p>So, for example, if a taxonomy has 9 items and MaxResults is set to 3, the screen displays only the first three items. When the site visitor clicks [Next], he sees items 4, 5 and 6, etc.</p> <hr/> <p>Important! If the EnablePaging property is set to True, the CacheInterval property is disabled.</p> <hr/>	Boolean

Property	Response	Data Type
EnableSearch	<p>Set to true if you want a search box (shown below) to appear above this taxonomy display. A site visitor can use the search to find content within a taxonomy/category that includes terms of interest.</p> <p>To suppress the search box, set to false.</p>	Boolean
		
FolderId	The ID of the folder from which to select data. To use the entire Ektron CMS400.NET , enter zero (0).	Integer
Hide	<p>Used to hide the directory control in design time and run time.</p> <p>True = Hide Sitemap trail</p> <p>False = Show Sitemap trail</p>	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	<p>Defines the way a link acts when it is clicked. The choices are:</p> <p>_blank - loads the link in a new blank window. This window is not named.</p> <p>_self - loads the link in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p>_parent - loads the link in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p>_top - loads the link in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>	String

Property	Response	Data Type
MarkupLanguage	<p>Identify the template markup file that controls the display of the taxonomy. For example, mytaxonomymarkup.ekml.</p> <p>If the .ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \[site_root]\workarea\customfiles\markup\mytaxonomymarkup.ekml.</p> <p>This property is blank by default. If nothing is entered, the following EkML file is used:</p> <p>\[site_root]\workarea\template\taxonomy\taxonomy.ekml</p> <p>See Also: "Ektron Markup Language" on page 754 and "taxonomy.ekml" on page 794</p> <p>If this property contains a valid EkML file or is left blank, the Displayxslt property value is ignored.</p>	String
MaxResults	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, set the EnablePaging property to true.</p> <p>If you do and more than the number of MaxResults are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="599 1099 1009 1326" data-label="Image"> <p>The screenshot shows a list of four names, each underlined and blue: Bill Lumbergh, Bob Porter, Bob Slydell, and Milton Waddams. Below the list are four navigation links: [First], [Previous], [Next], and [Last]. The [Previous] and [Next] links are highlighted in yellow.</p> </div>	Integer
Recursive	Set to True to include child folders of the parent folder.	Boolean

Property	Response	Data Type
ShowCount	<p>Indicates if the number of taxonomy items appears next to each category when displayed in the Web site. (See example below.)</p> <p>Default value is false.</p> <p>Breadcrumb: Top</p> <hr/> <p>Category: (What's This?)</p> <p>-Restaurant (13) -Business (1) -Transporta</p> <hr/> <p>Articles: (What's This?)</p>	Boolean
SuppressWrapperTags	<p>This property is set to <code>false</code> because Ajax uses <code><div></code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code>.</p>	Boolean
TaxonomyCols	<p>Enter the number of columns in which this taxonomy/category will appear on the page.</p>	Integer
TaxonomyDepth	<p>Enter the number of taxonomy levels to appear below each taxonomy/category. For example, if the taxonomy is Businesses > Restaurants > Pizza, and you set Taxonomy Depth to 2, only Business and Restaurants appears.</p> <p>The default value is 1.</p> <hr/> <p>Important! - For a live site, Ektron strongly recommends leaving this value at 1. Increasing this value can slow down your live Web server. However, for testing on a staging server, you can increase the depth.</p> <hr/> <p>A depth greater than 1 is only useful if you create a custom output using the Taxonomy's XML.</p>	Integer
TaxonomyId	<p>Enter the ID number of the taxonomy or category to appear in this server control. If you don't know the number, click the button and navigate to the taxonomy or category.</p> <p>When you select one, it appears in the center of the Visual Studio window.</p>	Integer

Property	Response	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

Discussion Board Server Controls

The Discussion Board feature provides a forum where topics can be discussed on your Web site. A site or membership user with Add Topic permission creates a topic and posts a starter question. Any site visitor can then reply.

Ektron, Inc. provides three server controls for use with Discussion Boards. The Forum server control, the ActiveTopics server controls and the PostHistory server control. Use the Forum server control to place a Discussion Board on a Web page. Use the ActiveTopics server control to display the most active topics and posts or the most recent topics and posts. Use the PostHistory server control to display a list of a given users posts.

- ["Forum Server Control" on page 147](#)
- ["ActiveTopics Server Control" on page 155](#)
- ["PostHistory Server Control" on page 157](#)

For more information about the Discussion Boards, see the **Ektron CMS400.NET** Administrator manual chapter "Discussion Boards." When using the Forum server control, be sure to follow the guidelines in the section "Inserting the Forum Server Control."

Forum Server Control

WARNING!

After placing a Forum server control on a page, follow the procedure described in "Updating the Page Command" on page 150. Otherwise, the user may get an error when posting a reply.


The Forum server control displays a discussion board on a Web page.

If your users are using the Javascript Editor for posts and replies, you can customize which buttons appear in editor.

Ektron Forums				
	Forum	Topics	Posts	Last Post
<u>Ektron Announcements</u>				
>>	<u>Product Announcements</u> Posts from Ektron about upcoming product releases	4	2	Tuesday, September 19, 2006 2:29 PM by BrianF
<u>Ektron Product Forums</u>				
>>	<u>CMS400NET</u> Discussions about CMS400.NET	373	828	Tuesday, October 10, 2006 8:57 AM by aromerodg
>>	<u>CMS300</u> Discussions about CMS300	3	5	Friday, August 25, 2006 8:25 AM by sk

Forum Server Control Properties

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Boolean
BoardID	The ID of the discussion board. If you don't know the ID, click the Ellipses button (), then sign in, browse to and select the discussion board. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42	Integer
DynamicParameter	Gets or sets the Querystring parameter to read a Forum ID dynamically. Set to "None - Use Default" if you want to always display the default Forum. None - Use Default - use the default Forum ID list. ID - reads a Forum's ID dynamically. ekfrm - reads a form block's ID dynamically.	String
ExcludeList	A comma separated list of custom user properties to exclude on the Forum's profile page. For additional information on custom user properties, see the Administrator Manual section "Managing Users and Permission" > "Custom User Properties."	String
Hide	Select False to display this server control on the page. Select True to suppress it.	Boolean

Property	Value	Data Type
HideUserProfile	<p>Set to True to suppress user profiles. The default is True. An Ektron CMS400.NET administrator can view a profile even if this property is set to true.</p> <p>True - hides user profiles</p> <p>False - displays user profiles. However, users or administrators can still suppress a profile via the Private Profile checkbox in the Forum Registration screen.</p> <p>For more information, see the Ektron CMS400.NET Administrator Manual section "Discussion Boards" > "Using Discussion Boards on Your Web Site" > "Suppressing User Information from the Forum Profile Display."</p>	Boolean
JavascriptEditorHeight	Set the height in pixels for the editor. The default is 400 .	Integer
JavascriptEditorHTMLMode	Set to True to allow users to edit their posts and replies in HTML. The default is False . See Also: "Allowing HTML Editing in the Javascript Editor" on page 151	String
JavascriptEditorToolbar	Decide which buttons are available in the Javascript Editor. See Also: "Adding and Removing Toolbar Buttons From the Javascript Editor" on page 151	String
JavascriptEditorWidth	Set the width in pixels for the editor. The default is 625 .	Integer
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
Theme	<p>Enter the custom theme's folder name. The theme can be relative to the site root or located in the current folder. For example:</p> <p>Relative - Theme=" /workarea/csslib/themes/winter"</p> <p>Current Folder - Theme="mytheme"</p> <p>If you do not specify a theme, the property uses the location defined by the discussion board's CSS theme property in the Workarea.</p> <p>See Also: "Using a Custom Theme" on page 150</p>	String

Property	Value	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <code><div></code> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Lets you use custom tag.</p>	String

Updating the Page Command

To prevent an error from appearing when a user posts a reply to the Web page that hosts the Forum server control, follow these steps.

1. Open the Web form onto which you inserted the Forum server control.
2. Access Source view.
3. Go to the top line of the Web form, which begins with `@ Page`.

```
<%@ PageLanguage="VB"AutoEventWireup="false" CodeFile="Default2.aspx.vb" Inherits="Default2" %>
```

4. Near the end of the line, enter `validaterequest=false`.
5. Build the page.

Using a Custom Theme

This property allows you to specify a custom location for your themes. By creating a custom location for your theme, you prevent it from being overwritten when you upgrade. If you do not specify a theme, the property defaults to the location defined by the discussion board's CSS theme property in the Workarea.

To create a custom theme, follow these steps.

1. Create a new subfolder on your site. In this example, we use a winter theme:
`<web root>/<your site>/Workarea/csslib/winter.`
2. Copy all files in the `/Workarea/Threadeddisc/themes` folder to the `winter` folder.
3. Change the image files to match your theme.

Tip! To find the name of an image, right click on it while viewing it in a browser. Then, select properties. The name appears at the top of the dialog box.

4. Change the threaded discussion's .css file to match your theme and save it with a custom name. You do not need to do this if an existing .css file meets your theme's needs.

IMPORTANT! Make sure the CSS theme property in the Workarea is pointing to the proper CSS file. See Also: The Administrator Manual section "Discussion Boards" >

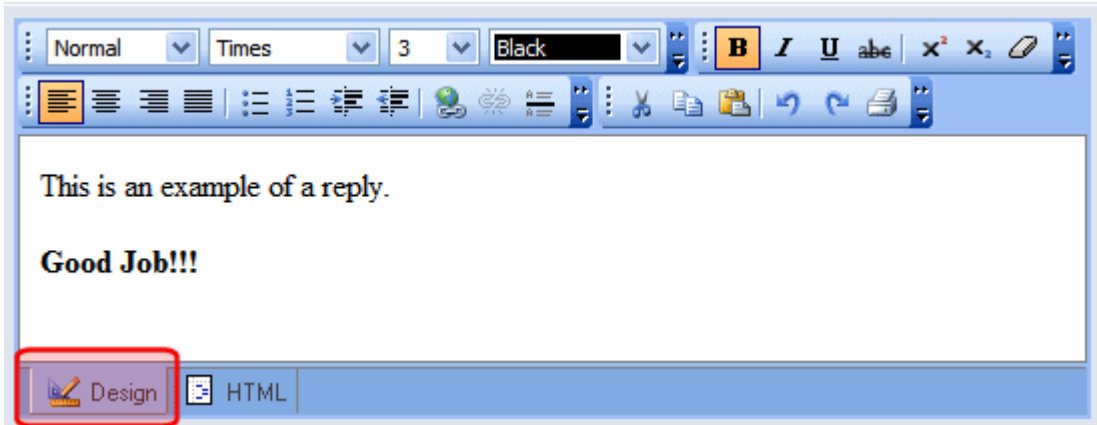
["Implementing Discussion Boards"](#) > ["Working with Discussion Boards and Categories"](#) > ["Creating a Discussion Board."](#)

5. View the discussion board in a browser to see the changes.

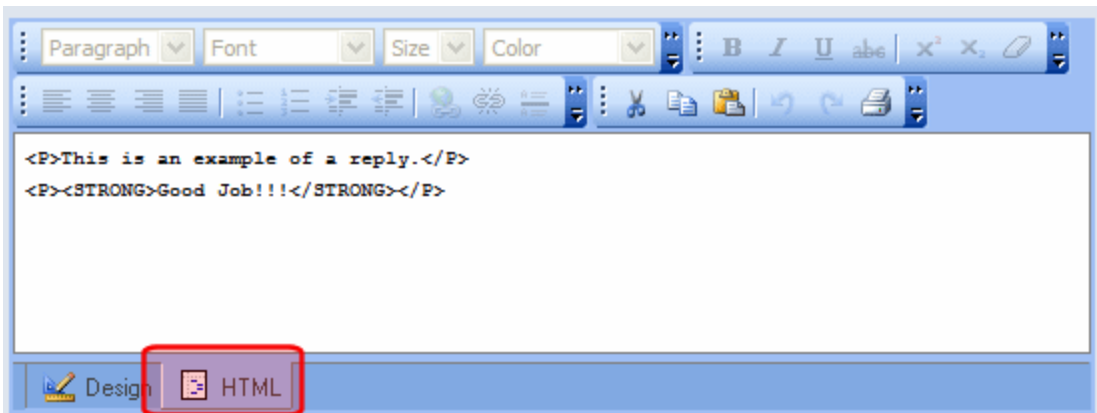
Allowing HTML Editing in the Javascript Editor

To allow Forum users to edit posts and replies in HTML mode, set the JavascriptEditorHTMLMode property to true. This creates tabs that allow the user to switch between Design view and HTML view.

Here is an example of a reply in Design view:



Here is the same reply in HTML view:



Adding and Removing Toolbar Buttons From the Javascript Editor

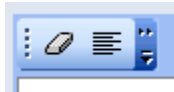

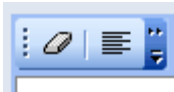
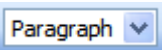
IMPORTANT!

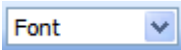

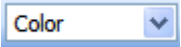











You can not create new buttons and add them to the Javascript Editor. You can only add and remove existing buttons.













Toolbar buttons on the Javascript Editor can be added and removed by editing JavascriptEditorToolbar property. The JavascriptEditorToolbar property contains a


series of string values that represent each button. Below is a list of buttons that are contained in the string for JavascriptEditorToolbar property.

The string values are separated by special characters. These special characters, a comma (,), vertical bar (|) and semicolon (;) allow a developer to decide on which toolbar a button appears and whether the buttons have a separator between them.

String Value	Button	Description
, (comma)	NA	<p>The , (comma) separates each string value on a toolbar.</p> <p>For example, place a , (comma) between the RemoveFormat string and the JustifyLeft string to have RemoveFormat button and JustifyLeft button appear on the same toolbar.</p> <p>Example Text: RemoveFormat,JustifyLeft</p> 
(vertical bar)	NA	<p>When placed between two string values, the (vertical bar) denotes the end of a toolbar and the beginning of the next. For example, if you place a (vertical bar) between the RemoveFormat string and the JustifyLeft string, the RemoveFormat button appears on the first toolbar and the JustifyLeft button appears on the second.</p> <p>Example Text: RemoveFormat JustifyLeft</p> 
; (semicolon)	NA	<p>When placed between two string values, the ; (semicolon) places a separator between two buttons. For example, if you place a ; (semicolon) between the RemoveFormat string and the JustifyLeft string, a separator appears between the RemoveFormat button and the JustifyLeft button.</p> <p>Example Text: RemoveFormat;JustifyLeft</p> 
ParagraphMenu		Change the heading size.

String Value	Button	Description
FontFacesMenu		Change the font style.
FontSizesMenu		Change the font size.
FontForeColorsMenu		Change the font color.
Bold		Make the text bold .
Italic		Make the text <i>italic</i> .
Underline		Under line the selected text. For example: <u>Here is some text</u>
Strikethrough		Apply strikethrough to selected text. For example: Here is some text
Superscript		Make selected text appear smaller and above text line. For example: The last word is ^{Superscript}
Subscript		Make selected text appear smaller and below text line. For example: The last word is _{Superscript}
RemoveFormat		Removes all formatting from the text.
JustifyLeft		Align paragraph so that it is arranged: • evenly on the left side (uneven on the right)
JustifyCenter		Align paragraph so that it is arranged: • in the center of each line
JustifyRight		Align paragraph so that it is arranged: • evenly on the right side (uneven on the left)
JustifyFull		Align paragraph so that it is arranged: • evenly on right and left side

String Value	Button	Description
BulletedList		Begin the line on which the cursor rests (or all selected lines) with a bullet (•).
NumberedList		Begin the line on which the cursor rests with a number. If the line above this line is: <ul style="list-style-type: none"> • not numbered, assign this line 1 • numbered, assign a number one more than the line above
Indent		Increase the current line's distance from the left margin.
Outdent		Decrease the current line's distance from the left margin.
CreateLink		Create a hyperlink and apply it to the selected text.
Unlink		Remove a hyperlink from the selected text.
InsertRule		Inserts a rule line between two groups of text.
Cut		Remove selected text and graphics. Place that data into temporary memory, also known as the "clipboard." (If you later cut or copy more information onto the clipboard, the new information overwrites the original information.)
Copy		Copy selected text and graphics into temporary memory. Leave selected data where it is. (If you later cut or copy more information into memory, the original information is lost.)
Paste		Insert the most recently cut or copied text and graphics at the current cursor location.
Undo		Reverse the most recent action, as if it never occurred.
Redo		Reverse the undo action.

String Value	Button	Description
Print		Prints the contents of the Javascript Editor.

ActiveTopics Server Control

The ActiveTopics server control displays either the most active topics for a forum or the most recent ones. The most active are determined by how many new posts are added to a topic or how many replies a post receives. The most recent are determined by the date of the post.


Below is an example of two ActiveTopics server controls. One control is set to Active, the other is set to Recent.

Most Active Topics	Most Recent Topics
<ul style="list-style-type: none"> • Cancer treatments (4/10/2006) • Welcome (10/11/2006) 	<ul style="list-style-type: none"> • Welcome (10/11/2006) • Cancer treatments (4/10/2006)

The items in the list are clickable links, which open to the topic.

ActiveTopics Server Control Properties

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

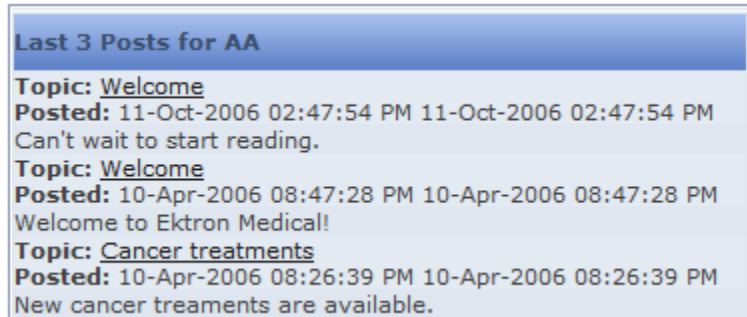
Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Boolean
BoardID	The ID of the board from which to get entries. If you don't know the ID, click the Ellipses button (), then sign in, browse to and select the discussion board. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String

Property	Value	Data Type
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: " Caching with Server Controls " on page 42	Integer
DisplayMode	Choose between Active or Recent. In Active mode, the server control displays the most active posts. In Recent mode, the server control displays the latest posts. The default is Active .	String
Hide	Select False to display this server control on the page. Select True to suppress it.	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
MaxNumber	The maximum number of topics listed. The default is 10 .	Integer
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
URLPath	Enter the path to the Forum server control's Web page. For example: <code>http://<your site>/siteroot/forum.aspx.</code> or <code>/CMS400Min/forum.aspx.</code> If your Forum page and your Active Topics page are in the same folder, you can just enter the name of the page. For example: <code>forum.aspx.</code>	String
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Lets you use custom tag.	String

PostHistory Server Control


The PostHistory server control displays a list of forum posts for a given user. The posts are displayed in order by date and contain the following:

- **Topic** - the topic that contains the post
- **Posted** - the date the post was created
- **Content** - the content within the post



PostHistory Server Control Properties

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Boolean
BoardID	The ID of the discussion board from which to get a user's posts. If you don't know the ID, click the Ellipses button (), then sign in, browse to and select the discussion board. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String

Property	Value	Data Type
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: " Caching with Server Controls " on page 42	Integer
Hide	Select False to display this server control on the page. Select True to suppress it.	Boolean
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
MaxNumber	The maximum number of posts listed. The default is 10 .	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
URLPath	The URL path to the page the where the Forum server control is located.	String
UserID	The ID of the user for whom to get the post history.	Integer
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The <code></code> tag is used to designate an in-line portion of an HTML document as a span element. Div - The <code><div></code> tag is used when you want to apply attributes to a block of code. Custom - Lets you use custom tag.	String

FormBlock Server Control

The FormBlock server control displays a content block associated with a form.

IMPORTANT!

If you create a template for an existing form content block, you must manually change its quicklink to point to the new template. This change does not occur automatically. This procedure is described in the Ektron CMS400.NET Administrator manual "Managing Library Assets" > "Updating Default Template for Multiple Quicklinks."

When added to a template and visited, the form content block might look like this. You can change the display to suit your needs by modifying its properties.

Absence Request Form

Name:

E-mail Address:

Position

Department:

Select Department ▼

Reason For Absence?

☐ Floating Holiday

☐ Vacation

☐ Sick Leave

☐ FMLA

☐ Other:

Dates of Absence:

to

Total Days Absent:

☐ With Pay

☐ Without Pay

Comments:

This subsection contains the following topics:

- ["FormBlock Server Control Properties" on page 160](#)
- ["Automatic versus Manual Generation of Form Tags" on page 161](#)

FormBlock Server Control Properties

The following table explains the properties of the FormBlock server control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
AddValidation	Set to True to add Validation to your main runat=server form.	Boolean
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.	Integer
DefaultFormID	The ID of a FormBlock that appears where you inserted this server control if no other form block is identified. If you don't know the ID number of the form block, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.	Integer
DynamicParameter	To make this form block dynamic, select id . When you do, this server control uses the form block passed as a URL parameter.	String

Property	Value	Data Type
Fields	Displays a list of fields that are defined in the form. These fields are read only. This is an excellent way of displaying the field names used on the form. With this list of names, you can create events using the fields without having to enter the Workarea to see the names.	
Hide	Used to hide a form block in design time and run time. True = Hide form block False = Show form block	Boolean
IncludeTags	Determines if tags are generated automatically of manually. See Also: "Automatic versus Manual Generation of Form Tags" on page 161	Boolean
Language	Set a language for viewing form content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

Automatic versus Manual Generation of Form Tags

When using the FormBlock Server control's Include tags property, you have two options.

Option	Recommended if you are developing	How to Enable
Let Ektron CMS400.NET generate form tags automatically	A pure script .aspx page	Set Include Tags property to True .

Option	Recommended if you are developing	How to Enable
Modify HTML form tags in the .aspx file	An .aspx page and associated code-behind web form	<p>Set Include Tags property to False.</p> <p>Here is the default .NET generated form tag:</p> <pre><form id="Form1" method="post" runat="server"></pre> <p>Modify the form tag as indicated in red:</p> <pre><form id="Form1" method="post" runat="server" OnSubmit="EkFmValidate(this);"></pre>

HTMLEditor Server Control

Introduction

This server control provides the ability to edit CMS content. The eWebEditPro editor can easily be placed onto a page for content editing. There is flexibility to control the toolbars, the available functionality, and the operations of the editor.

WARNING!

After placing an HTMLEditor server control on a page, follow the procedure described in "Updating the Page Command" on page 150. Otherwise, the user may get an error when submitting the content.

The control, at run time, produces the fields listed below. They are posted with the content and can be processed in the receiving page.

- **[ControlName]_Action** = the action to take - save, check in, undo checkout
- **[ControlName]_ContentTitle** = the content title
- **[ControlName]_ContentId** = the ID of the content
- **[ControlName]_Language** = the language of the content

The content title caption is surrounded with span tags that have this ID:

- **[ControlName]_TitleLabel**

All information for all content information fields (except the content itself) is surrounded with a div tag with this ID:

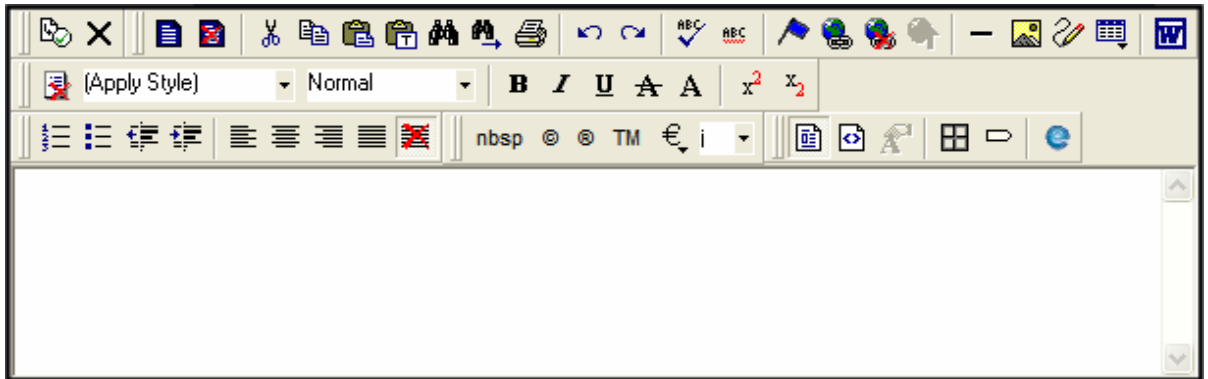
- **[ControlName]_ContentInfo**

The valid values for the [ControlName]_Action value are:

- **Publish** - submit the posted content for publication
- **CheckIn** - check in the content
- **Save** - save the content and redisplay in the editor
- **UndoCheckout** - undo the checkout of the content

The server control displays the Ektron CMS400 HTML editor on a Web page. By adding the HTMLEditor to a Web page, site visitors can create or edit HTML

content for a site or project. Below is an example of the editor with the default toolbars.



The HTML Editor server has two modes.

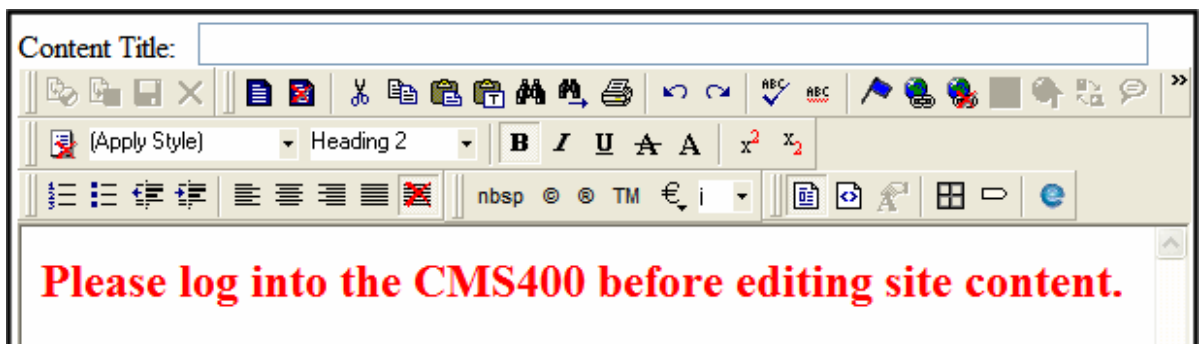
- working with Ektron CMS400.NET - See ["Working with Ektron CMS400.NET Content" on page 164](#)
- working as an independent Html editor - See ["Working with Non Ektron CMS400.NET Content" on page 165](#)

Working with Ektron CMS400.NET Content

When you work with Ektron CMS400.NET, site visitors need to be logged in. When logged in, a site visitor can edit existing content or create new content.

To edit existing content, the content's numeric ID must be passed to the ContentId property. If you enter an ID in the ContentId property, the HTML Editor assumes you are working with CMS400.NET. To create new content, leave ContentId blank.

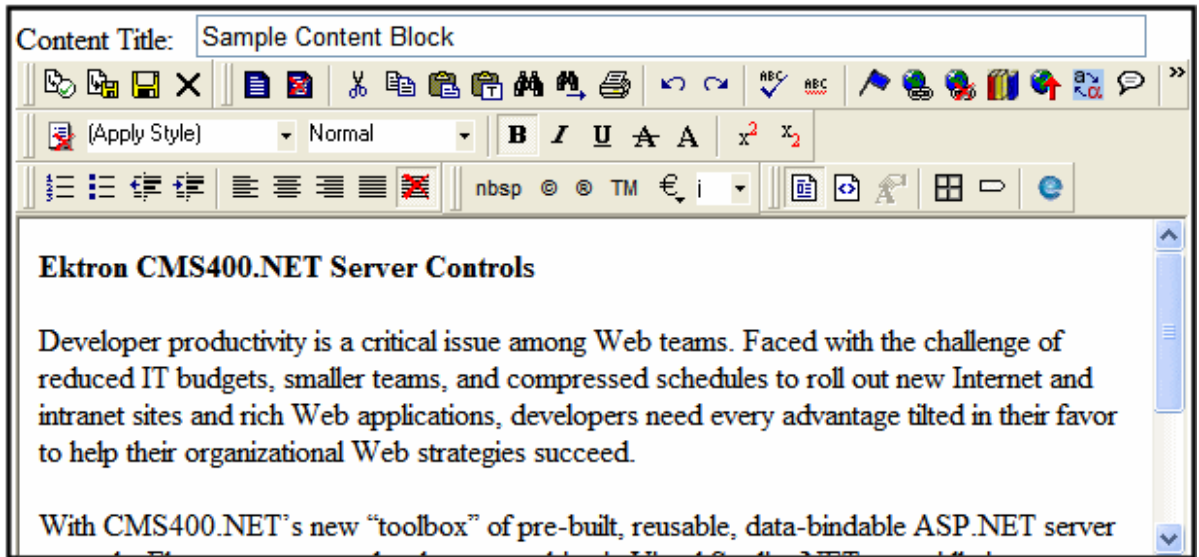
When content is loaded into the editor but the site visitor is not logged in, the editor displays the following message:



Notice the CMS toolbar's Publish, Check-in, Save, and Cancel buttons are grayed out.



Once a site visitor logs in, the content loads and the CMS toolbar is active.



Working with Non Ektron CMS400.NET Content

If a user is not logged to Ektron CMS400.NET, the HTML editor is used as an independent editor. For examples of integrating the Html Editor with your Web site, see the following files in `\webroot\siteroot\Developer\ContentBlock`.

- **EditAnyHtmlSample.aspx.vb** - shows an example when not logged in
- **EditHtmlSample.aspx.vb** - shows an example when logged in

Uploading Files with an Action Page

When working with non-CMS400.NET content, create an action page to handle the uploading of files. In addition, edit the `cms_config.aspx` file to show the location of the new action page. To edit `cms_config.aspx`, follow these steps:

1. Open `cms_config.aspx`, located in `site root\Workarea\ewebeditpro\`.
2. Between the `<Mediafiles>` tags, find `<transport allowupload="true" type="ftp" xfer="binary" pasv="true">`
3. Change the `Type` element's value from `ftp` to the location of your action page. For example: `Type="CMS400Min/myfileselect.aspx"`
4. Save and close the file.

HTML Editor Server Control Properties

The following table explains the properties of the HTML Editor server control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Archived	Signifies that the selected content is archived. For more information on archived content, see the User Manual section "Scheduling Content to Begin and End" > "Setting Archive Options".	Boolean
Authenticated	A read-only property that indicates if you are logged in to the CMS Explorer and can use it to browse to content, collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.	String
CmsSitePath	DESIGN TIME ONLY: Used to find the web service required to access the CMS in Design Time.	String
Content ID	Specifies the content ID to load. This value must match a value that exists in the CMS database. The selection of content is similar to that in the ContentBlock control. The content can only be accessed if the user is logged in. This value can be set only on editor load, before the editor and contents are rendered. After editor load, this value is ignored. If this value is 0, then the code behind must handle making the calls that check out content and setting the content at edit time by placing it into the .Text property for the editor. A hidden field, named EktContentID_[ControlName], is created to hold this value. It is posted up with the rest of the content.	Integer
ContentType	This value describes the type of content the selected ID refers to. If this value is blank, then the content is a custom type, to be defined type, or unsupported type. These are the possible types: HTML Form	String

Property	Value	Data Type
EditorReadyCall	<p>This specifies which JavaScript function to run when the HTML editor is ready to start processing commands. This happens after the initialization of the editor and the loading of the initial content.</p> <p>This is the name, not the full function definition, of a JavaScript function. The function must follow the format defined in the eWebEditPro documentation for this type of method. Below is an example:</p> <pre>function initEditorIsReady(sEditor, strURL, strAutoURL) { ... }</pre> <p>where the "initEditorIsReady" string is given to this property. If this string is empty, the default CMS method is called.</p>	String
EnableUpload	<p>If this is 'true' then the user can upload locally selected images and linked files that they have put into their content. If enabled, the automatic upload mechanism will allow upload the local files automatically with minimal user intervention. If this value is 'false' then the site developer must provide an upload page, such as the CMS library, where the upload can occur.</p>	Boolean
IncludeContentInfo	<p>If true, this will include the information about the content in fields posted with a form post. (This fields are defined in the Introduction of this section.)</p> <p>Set the value to 'false' if the page will maintain these values or the page is just using the editor to edit general formatted text.</p> <p>If the information must be posted, but not shown to the end user, then define the class [ControlName]_TitleLabel as having no display, or set the ShowTitle property to false.</p>	Boolean
Language	<p>Specifies the ID of the language to use for the content to load. This value must match a value that exists in the CMS database.</p>	Integer
SafeEncode	<p>Decide whether to encode the HTML content.</p>	Boolean

Property	Value	Data Type
ShowCMSToolbar	Controls whether the CMS toolbar is displayed in the editor's toolbar. If this value is true, then commands containing the CMS functionality are displayed in the editor's toolbar. The commands are for check-in, publish, and edit cancellation. <i>See Also:</i> "ShowCMSToolbar" on page 169	Boolean
StyleSheet	Assigns a single sheet to the edited content.	String
Text	The HTML text being edited.	String
ToolbarLevel	This specifies the level of the toolbar feature set offered to the user. This defines the amount of functionality offered to the user. <i>See Also:</i> "ToolbarLevel" on page 169 .	String
ToolbarResetCall	<p>This specifies which JavaScript function to run when the toolbar resets. The toolbar reset offers the opportunity for the script to affect the operation of the toolbar with actions such as adding or removing commands, disabling commands, or creating entirely new toolbars. This is the name, not the full function definition, of a JavaScript function. The function must follow the format defined in the eWebEditPro documentation for this type of method. Below is an example:</p> <pre>function initToolbarCommands(sEditor, strURL, strAutoURL) { ... }</pre> <p>where the "initToolbarCommands" string is given to this property.</p> <p>A toolbar reset is only called when the toolbar actually resets. This often happens when the editor starts, however, if the user has been given the opportunity to save their own toolbar setup then this will not be called on startup. It is always called when the user selects to reset the toolbar.</p> <p>If this string is empty, the default CMS method is called.</p>	String
TranslationFile	The translation file (or stream) that contains the XML elements for the translation of strings used in the editor. If this is blank, the client system's windows setting is used to select from the series of default translation files.	String

ShowCMSToolbar

NOTE For information on the Toolbar buttons and there functions, see the User Manual Section > "Toolbars".

The CMS Toolbar appears when the ShowCMSToolbar property is set to true. There are two versions of the toolbar that appear:

- site visitor is logged into Ektron CMS400.NET



- site visitor is not logged in

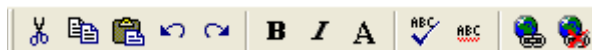


ToolbarLevel

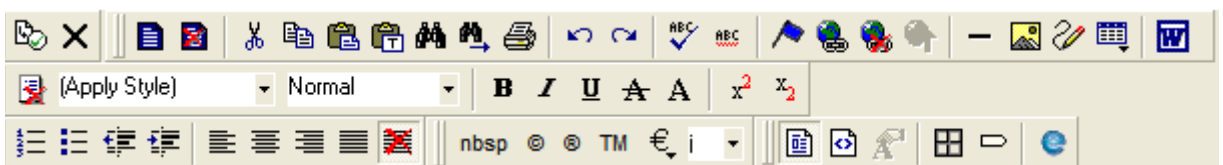
NOTE For information on the Toolbar buttons and there functions, see the User Manual Section > "Toolbars".

This property specifies the level of the toolbar feature set offered to site visitors. It may be that they need a minimal set of functionality to keep the editing simple. Or, they need full functionality for a major edit. The choices available are:

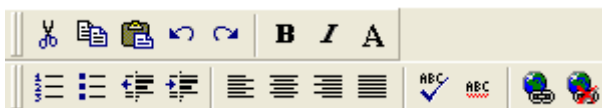
- **None** - No toolbars are shown.
- **Reduced** - Allows for minimal changes to the text.



- **Default** - Loads the toolbars typically need for working with content.

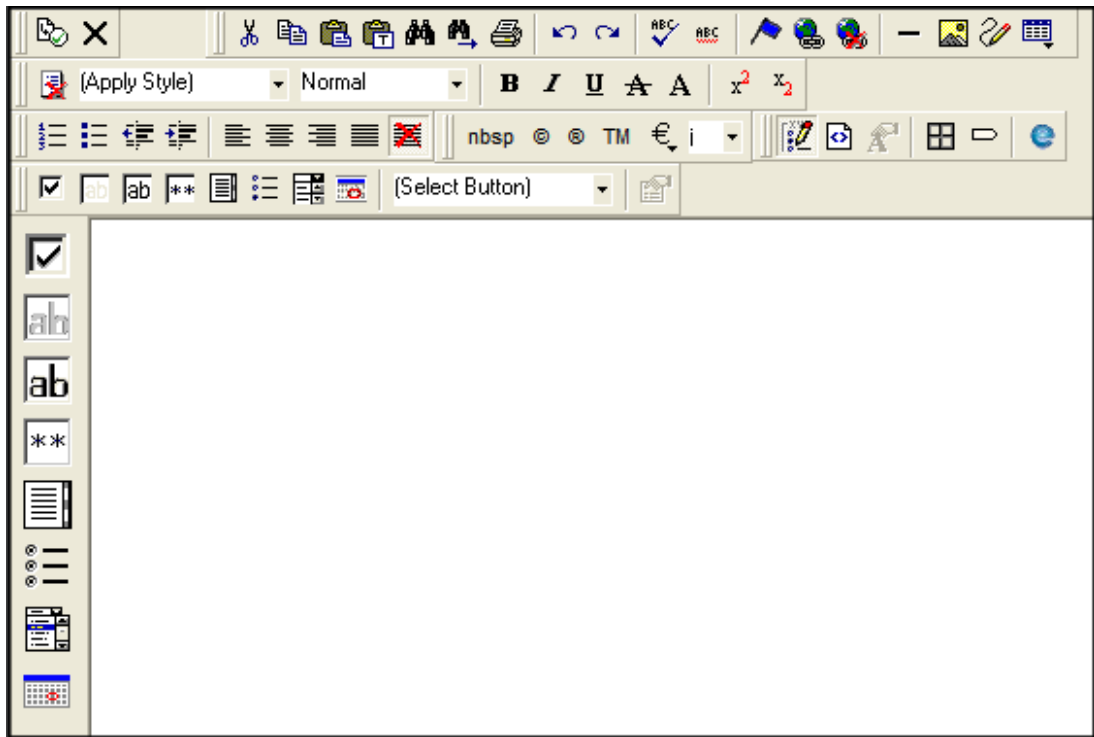


- **Calendar** - Loads the Paragraph and Paragraph Format toolbars.

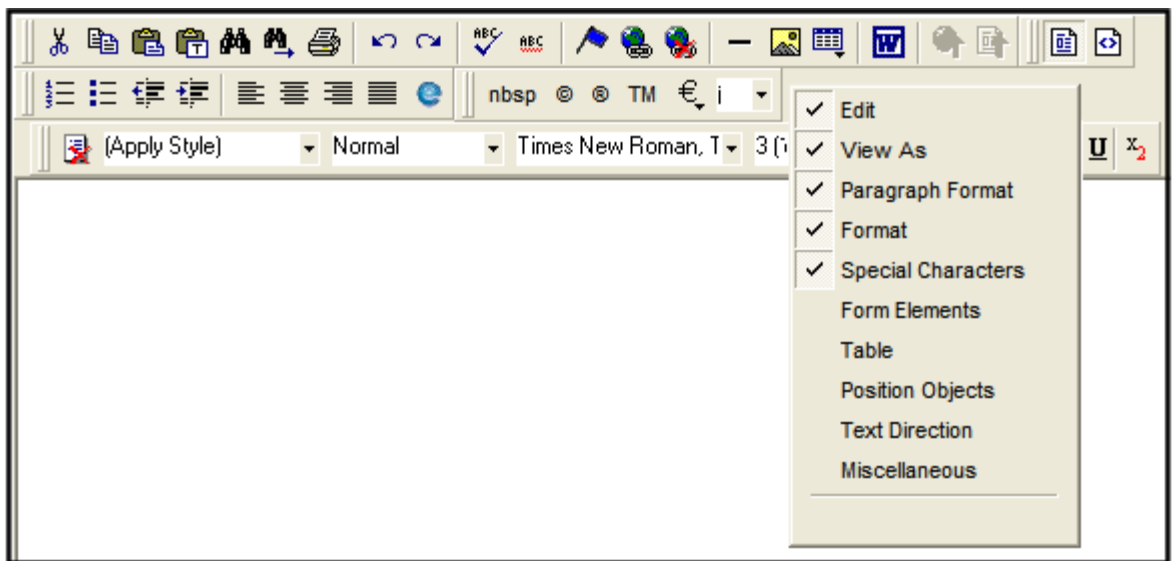


- **FormDesign** - Loads the toolbars that are needed to work with forms.

IMPORTANT! The Data Designer is not supported when using the HTML editor server control.

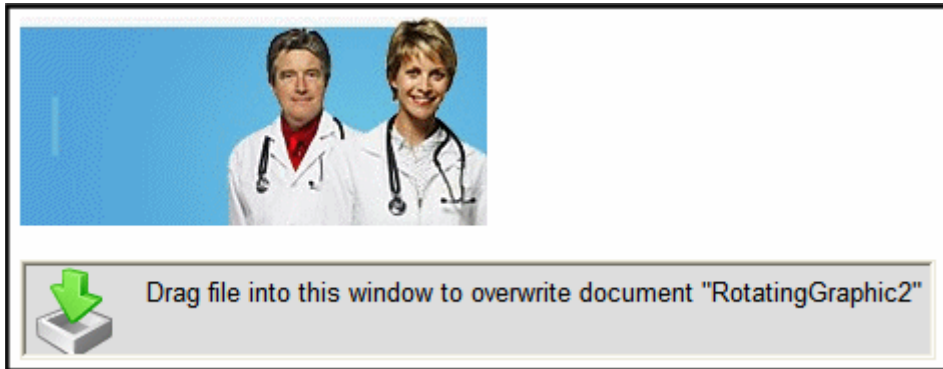


- **Custom** - Allows a site user to define the toolbars that appear by right clicking on the toolbar area of the screen.



ImageControl Server Control

The ImageControl server control, when viewed on a Web form, displays an image stored within the CMS400.NET document management feature. When a user is logged in and has permission to edit the image, he can right click the image and select edit. This creates a drag and drop box that the user can use to update the file.



IMPORTANT!

Images are stored as assets using the Document Management feature. This control does not use the CMS400.NET Library.

This section contains the following topics:

- ["ImageControl Server Control Properties" on page 171](#)

ImageControl Server Control Properties

The ImageControl server control properties are described in this table.

NOTE

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String

Property	Description	Data Type
DefaultImageID	The image asset's content ID you want to display. If you don't know the ID number of the asset, use the CMS Explorer to browse to it. See A/so: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
DynamicParameter	Gets or sets the Querystring parameter to read an image asset's ID dynamically. To have the default image ID used, leave blank.	String
FolderID	The ID of the folder where images are added. If you don't know the ID number of the folder, use the CMS Explorer to browse to it. See A/so: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	Integer
Hide	Used to hide output of AssetControl in design time and run time. True = Hide AssetControl False = Display AssetControl	Boolean
Language	Set a language for viewing the collection. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an inline portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

LanguageAPI Server Control

The LanguageAPI server control lets a developer force a particular language for a Web site. You can do this simply by dropping the server control on the page and then choosing a language in the SiteLanguage property box. You can also override site language logic by programmatically using the LanguageAPI server control to detect the browser's language, and display the site in that language.

This subsection contains the following topics:

- ["LanguageSelect Server Control vs. LanguageAPI Server Control" on page 173](#)
- ["LanguageAPI Server Control Properties" on page 173](#)
- ["Code Behind Only Properties and Methods" on page 174](#)

LanguageSelect Server Control vs. LanguageAPI Server Control

The LanguageSelect server control lets users view your site in the language they choose. The LanguageAPI server control, on the other hand, forces users to view a site in a particular language. For information on LanguageSelect, see ["LanguageSelect Server Control" on page 176](#).

LanguageAPI Server Control Properties

The LanguageAPI server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
SiteLanguage	Sets the site language. Runs through the IsValid function to verify that the language is an active language in the system.	String
SiteLanguageID	Numeric value of the site language. This is the property you will use if you are using code behind to set the sites language ID. Example: 1036 = French For a list of supported languages, see The Administrator Manual > "Appendix A: List of Supported Languages"	Integer

Code Behind Only Properties and Methods

The following is a list of LanguageAPI specific properties and methods that can only be used programmatically.

Property	Value	Data Type
CurrentLanguage	Read only. Returns the current language.	String
CurrentLanguageID	Read only. Returns the current language ID.	Integer
DefaultLanguage	Read only. What the default language of the site is set to. For example, the demo site is "English (Standard)".	String
DefaultLanguageID	Read only. Returns the value that is the default language ID of the site. For example, the demo site is "1033" for English.	Integer
GetLanguage	This method returns a string. Pass in a valid language ID and it will return the language name.	Integer Argument
GetLanguageID	This method returns an integer. Pass in a valid language name and it will return the language ID.	String Argument
IsValid	This method returns a boolean. You can pass in a language ID or a string and it will tell you if the system is supporting it.	Argument
LanguageIDList	Read only. Lists all the language IDs that are activated in Ektron CMS400.NET . For information on how to enable languages, see The Administrator Manual > "Multi-Language Support" > "Working with Multi-Language Content" > "Determining Which Languages are Available".	Array of Integers
LanguageTitleList	Read only. Lists all the languages that are activated in Ektron CMS400.NET .	Array of Strings

Property	Value	Data Type
MultiLanguageEnabled	Read only. Tells if the site supports multi-language mode. True = Multi-language enabled False = Multi-language not enabled Multi-language is enable by default in Ektron CMS400.NET . To disable Multi-language, see Administrator Manual > "Multi-Language Support" > "Working with Multi-Language Content" > "Disabling Support for Multiple Language Content".	Boolean

Using the LanguageAPI Server Control Programmatically

This example uses a logo that is not managed through Ektron CMS400.NET. It retrieves the current language from the LanguageAPI control, and uses that information to choose the logo version to display. The codebehind looks like this.

```
Select Case LanguageAPI1.CurrentLanguageID

    Case 1031

        Image1.ImageUrl = "germanlogo.gif"

    Case 1033

        Image1.ImageUrl = "englishlogo.gif"

    Case 1036

        Image1.ImageUrl = "frenchlogo.gif"

End Select
```

LanguageSelect Server Control

The LanguageSelect server control displays a language selection dropdown list on an Ektron CMS400.NET Web page.

This control lets a site visitor select a language in which to view the site. Here is what the control looks like when published on a Web page.



It lists all languages selected on the **Settings > Configuration > Language settings** screen. (For more information, see the "Multi-Language Support" chapter of the **Ektron CMS400.NET Administrator** manual.)

You can place this control in any location of any page on your site. On the sample site, it appears in the lower left corner of the home page.

This subsection contains the following topics:

- ["LanguageSelect Server Control vs. LanguageAPI Server Control" on page 176](#)
- ["LanguageSelect Server Control Properties" on page 176](#)

LanguageSelect Server Control vs. LanguageAPI Server Control

The LanguageSelect server control lets users view your site in the language they choose. The LanguageAPI server control, on the other hand, forces users to view a site in a particular language. For information on LanguageAPI, see ["LanguageAPI Server Control" on page 173](#).

LanguageSelect Server Control Properties

The following table explains the properties of the LanguageSelect control.

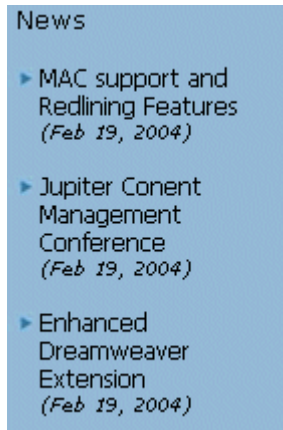
NOTE	The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.
-------------	--

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
Hide	Used to hide language select dropdown box and label in design time and run time. True = Hide language select dropdown box and label False = Show language select dropdown box and label	Boolean
LabelName	Lets you define the label next to the language select dropdown box.	String
Language	Set a language for the Language Select Box. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is False . True - Suppress wrap tags. False - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is Span . Span - The tag is used to designate an in-line portion of an HTML document as a span element. Div - The <div> tag is used when you want to apply attributes to a block of code. Custom - Allows you to use a custom tag.	String

ListSummary Server Control

The ListSummary server control is used to display, on a Web page, a list of content blocks in a selected folder. Optionally, the display can include content in all subfolders of the selected folder.

When added to a template and visited, a List Summary looks like this.



It displays each content block's title and summary information. You can modify the display to suit your needs by modifying its properties.

This subsection contains the following topics:

- ["ListSummary Server Control Properties" on page 178](#)
- ["OrderKey Property" on page 184](#)
- ["Retrieving the XML Structure of a List Summary" on page 185](#)

NOTE In contrast to a List Summary, a ContentList server control displays selected content items from any CMS400.NET folder. See Also: ["ContentList Server Control" on page 117](#)

ListSummary Server Control Properties

The ListSummary server control properties are described in this table.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to the folder needed for the FolderID property. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15	String
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42</p> <hr/> <p>Important! If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Integer
ContentParameter	Checks the QueryString for this value and replaces the list summary with a content block when specified. Leave blank to always display the list summary.	String
ContentType	<p>Determines the type of content that appears in the list summary. The default is <code>Content</code>.</p> <p>AllTypes - displays all of the content types for the given folder.</p> <p>Content - displays a list of content blocks.</p> <p>Forms - forms appear in the list summary.</p> <p>Archive_Content - archived content blocks appear in the list summary.</p> <p>Archive_Forms - archived forms appear in the list summary.</p> <p>Assets - assets, such as offices documents, appear in the list summary.</p> <p>Archive_Assets - archived assets appear in the list summary.</p> <p>LibraryItem - library items appear in the list summary.</p> <p>Multimedia - multimedia items appear in the list summary.</p> <p>NonLibraryContent - all types of content appear in the list summary except for library items.</p> <p>ImageContent - images appear in the list summary.</p> <p>DiscussionTopic - forum topics appear in the list summary.</p>	String

Property	Description	Data Type
Direction	<p>How to order the hyperlinks on the list.</p> <p>The sort field is determined by the OrderKey property.</p> <p>ascending - they are arranged A, B, C or 1,2,3.</p> <p>descending - they are arranged. Z,Y,X or 3,2,1</p> <p>If sorting by date, descending puts the most recent first.</p>	either ascending or descending
DisplayXslt	<p>Determines how the information on the page is displayed</p> <p>None-databind only</p> <p>ecmNavigation - lists the title of every content block in the folder</p> <p>See Also: "Example of ecmNavigation Display" on page 101</p> <p>ecmTeaser - lists the title of every content block in the folder plus the content summary</p> <p>See Also: "Example of ecmTeaser Display" on page 102</p> <p>Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p>Important! If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String

Property	Description	Data Type
EnablePaging	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of List Summary items while controlling the amount of screen space. To accomplish this, the List Summary display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you set this property to true, and the number of List Summary items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="594 542 1002 768" data-label="Image"> </div> <p>So, for example, if a List Summary has 9 items and the <code>MaxResults</code> property is set to 3, the screen displays only the first three items. When the site visitor clicks [Next], he sees items 4, 5 and 6, etc.</p> <p>True = Use paging feature False = Ignore paging feature</p> <hr/> <p>Important! If the <code>EnablePaging</code> property is set to True, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Boolean
FolderID	<p>The folder that contains the content blocks which appear in the list summary. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p> <p>The <code>Recursive</code> property determines whether content blocks in this folder's child folders also appear.</p>	Integer
GetHtml	<p>Set to True if you want to retrieve and display content (that is, the HTML body) for all content in the list summary. For example, to display content inside a Web server control such as a GridView.</p> <p>True = Get and display HTML for each content block in the list summary False = Do not get and display HTML.</p>	Boolean

Property	Description	Data Type
Hide	Used to hide output of the list summary in design time and run time. True = Hide results False = show results	Boolean
Language	Set a language for viewing the list summary. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	Defines the way a link acts when a link is clicked. _blank - This target causes the link to always be loaded in a new blank window. This window is not named. _self - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target. _parent - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent. _top - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.	String
MarkupLanguage	Identify the template markup file that controls the display of the list summary. For example, mylistsummary.ekml. If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \\cms400Min\workarea\customfiles\markup\mylistsummary.ekml. See Also: "Ektron Markup Language" on page 754 and "listsummary.ekml" on page 783 If you enter a valid EkML file, the Displayxslt property value is ignored.	String

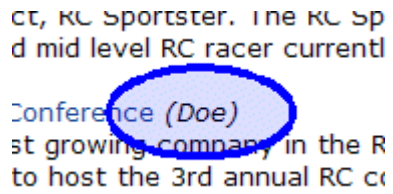
Property	Description	Data Type
MaxResults	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter 0 (zero).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, set the <code>EnablePaging</code> property to true.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="594 584 1002 811"> <p>The screenshot shows a list of four names: Bill Lumbergh, Bob Porter, Bob Slydell, and Milton Waddams. Below the list are four navigation links: [First], [Previous], [Next], and [Last]. The links are styled with a light blue background and a thin border.</p> </div>	Integer
OrderKey	<p>Sort the list by one of the values.</p> <p>Title</p> <p>DateModified</p> <p>DateCreated</p> <p>LastEditorFname</p> <p>LastEditorLname</p> <p>See Also: "OrderKey Property" on page 184</p>	String (must be one of the values)
Random	<p>Set to True if you want to randomly display one content block in the specified folder. The content changes each time a user views the page.</p> <p>True - randomly display one content block.</p> <p>False - display the list summary normally.</p>	Boolean
Recursive	<p>Determines if the display includes content blocks in child folders of the selected folder.</p> <p>True - include content blocks from child folders.</p> <p>False - do not include content blocks from child folders.</p>	Boolean
SuppressWrapperTags	<p>This property is set to <code>false</code> because Ajax uses <code><div></code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code>.</p>	Boolean

Property	Description	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The tag is used to designate an inline portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

OrderKey Property

The OrderKey property determines what information appears to the right of the hyperlinked title. The values are described below.

Value	Description	Web Page View
DateModified	Displays the date the content was last modified.	<p>, RC Sportster. The RC Sportster is design mid level RC racer currently on the market</p> <p>ference (8/27/2002 2:44:44 PM)</p> <p>growing company in the RC</p> <p>host the 3rd annual RC conference in Atl</p>
DateCreated	Displays the date when the content was first created.	<p>id level RC racer currently on the mark</p> <p>erence (4/25/2002 4:42:55 PM)</p> <p>owing company in the RC</p> <p>ost the 3rd annual RC conference in A</p>
LastEditorFname	Displays the first name of the user who last edited the content block.	<p>mid level RC racer currentl</p> <p>ference (John)</p> <p>growing company in the R</p> <p>host the 3rd annual RC cc</p>

Value	Description	Web Page View
LastEditorLNames	Displays the last name of the user who last edited the content block.	

Retrieving the XML Structure of a List Summary

Retrieving the XML structure of XML content allows for greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a ListSummary server control onto it.
3. Set the FolderID property.
4. Drag and drop a Textbox on the Web form.
5. Set the TextMode property to MultiLine.


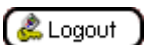
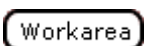
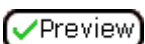

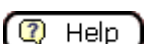
NOTE Ektron recommends setting the width of the text box to at least 400px.

6. On the code behind page, add the following line.

```
Textbox1.Text = ListSummary1.XmlDoc.InnerXml
```
7. Build the project.
8. View the Web form in a browser.
9. The list summary's XML structure appears in the textbox.

Login Server Control

The Login server control paints a login button on the template when displayed in a browser. When the Login server control is inserted and the project is built, the control displays the following buttons on a Web page.

Button	Description
 Login	When user is not logged in, this button appears. Clicking the button opens the login window, where a user can enter a username and password. Upon authentication, the user is logged in to the Ektron CMS400 Web site.
 Logout	After a user logs in, this button replaces the login button to let the user log out.
 Workarea	When logged in, this button appears under the logout button, allowing the user to access the Workarea.
 Preview	Lets the user preview the entire Web site as if all checked-in content were published. For more information, see the Ektron CMS400.NET User Manual section "Logging In and Out" > "Site Preview."
 X Preview	Turns off site preview mode.
 Help	Launches online, context sensitive help for Ektron CMS400.NET .

This subsection contains the following topics:

- ["Placing a Login Button" on page 186](#)
- ["Login Server Control Properties" on page 187](#)

Placing a Login Button

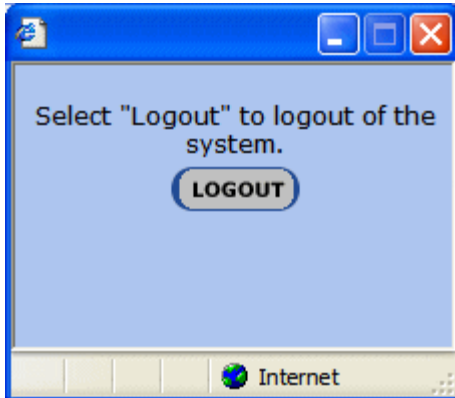
You can add any number of login buttons to a template. You can insert a login button on each template, or set up a special Web page, called login.aspx, from which users can log into the CMS400 site without the public being able to access the page.

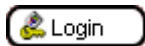
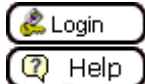
Login Server Control Properties

The following table explains the properties of the Login server control.

NOTE The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	String
AutoAddType	Set this type to define automatically added users as membership users or CMS400.NET content authors. Author = CMS400.NET content authors Member = site visitor membership users	String
AutoLogin	When set to true, users are automatically logged in using Active Directory authentication. They do not need to enter a username or password. The default is False. True = Use Active Directory authentication when logging in. False = Do not use Active Directory authentication when logging in. Important! For this property to function properly, you must be using Active Directory authentication with your CMS400.NET Site.	Boolean
Hide	Used to hide login button in design time and run time. True = Hide login button False = Show login button	Boolean
Language	Set a language for the Login server control. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer

Property	Value	Data Type
OnlyAllowMemberLogin	<p>This changes the Login button so only membership users can log in. This prevents someone from trying to login as an Ektron CMS400.NET user and gain access to the Ektron CMS400.NET Workarea. If a CMS Author tries to log in using this dialog box, a message appears in the box:</p> <p>"Only members are allowed to login here."</p> <p>The default is False.</p> <p>True = Allow only membership users to login.</p> <p>False = Allow Ektron CMS400.NET users and membership users to login.</p>	Boolean
PromptLogout	<p>When set to False, the logout process will not include the Logout Prompt window (shown below).</p>  <p>True = Users must click the Logout button in the Logout prompt window before they are logged out.</p> <p>False = Users are logged out without having to use the Logout prompt window.</p>	Boolean

Property	Value	Data Type
SuppressHelpButton	<p>Hides the Help button that appears below the Login button when set to true. The default is False.</p> <p>True = Do not display the Help button.</p>  <p>False = Display Help button.</p>  <p>If you are editing this server control from a text file and want to suppress the Help button, add the following code to the login tag source:</p> <pre><CMS:Login ID="Login1" runat="server" SuppressHelpButton="True"/></pre>	Boolean
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>	Boolean
WrapTag	<p>Allows a developer to specify a server control's tag. The default is Span.</p> <p>Span - The tag is used to designate an in-line portion of an HTML document as a span element.</p> <p>Div - The <div> tag is used when you want to apply attributes to a block of code.</p> <p>Custom - Allows you to use a custom tag.</p>	String

Map Server Control

Introduction

The Map server control displays a map that flags locations of interest to your site visitors. Each location is a CMS content item to which map information was added. For example, if your site hosts a school district, each map location could represent one school in the district.

A visitor can zoom the map in and out, get directions to any location, and narrow the list of locations using a text search. For example, if your map initially flags all schools in a geographic area, the visitor can redraw the map to show only schools with a gym.

If you want the map to show events, you can apply dates to **Ektron CMS400.NET** content. If you do, a site visitor using the map can search by date as well as location to find events of interest.

Search
Find What
Direction
☐ Hide zoom control

Road
Aerial

Results 1 - 6 of 6

No.	Title
1.	Amherst House of Pizza 131 State Route 101A # 6 Amherst, NH 603) 886-5543
2.	Domino's Pizza 556 Nashua St, Milford, NH (603) 673-2700
3.	Pizza Top 183 Elm St, Milford, NH (603) 673-0037
4.	You You Japanese Bistro 150 Broad St #4, Nashua, 03063 Japanese, Korean, EurAsia
5.	Budkley's Steak House 438 Daniel Webster Highw Merrimack, NH 03054 Steak wood grilled to perf
6.	Surf Restaurants

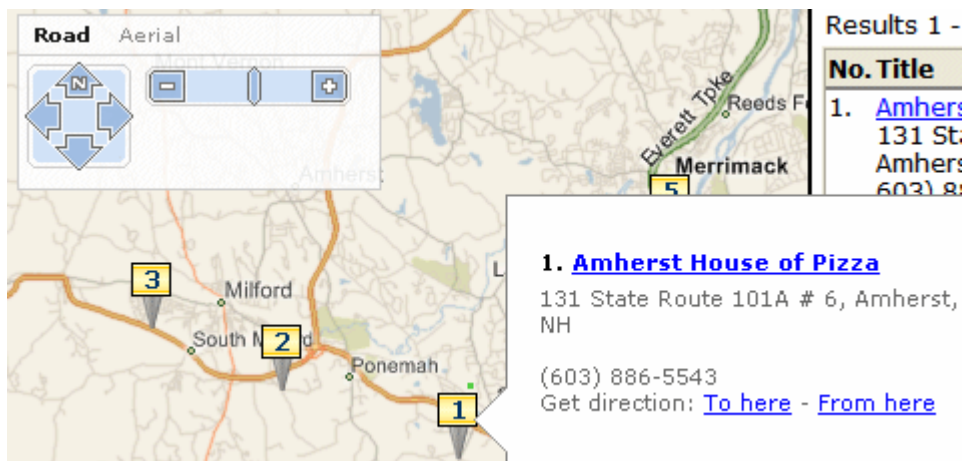
IMPORTANT!

As a map's location boundaries change, only locations within the boundaries appear. Likewise, if a date is assigned to content, only content within the selected date range appears.

If a map has at least one flagged location, a box can appear to its right with information about each flagged location. The locations are sorted by distance to starting location. From this box, you can

- view the content to which the location is assigned
- find the distance to a location
- center the map on a location
- get directions to a location

If you click any flagged location, a box appears showing a link to that content in **Ektron CMS400.NET**, additional information about the item, and a link to directions.



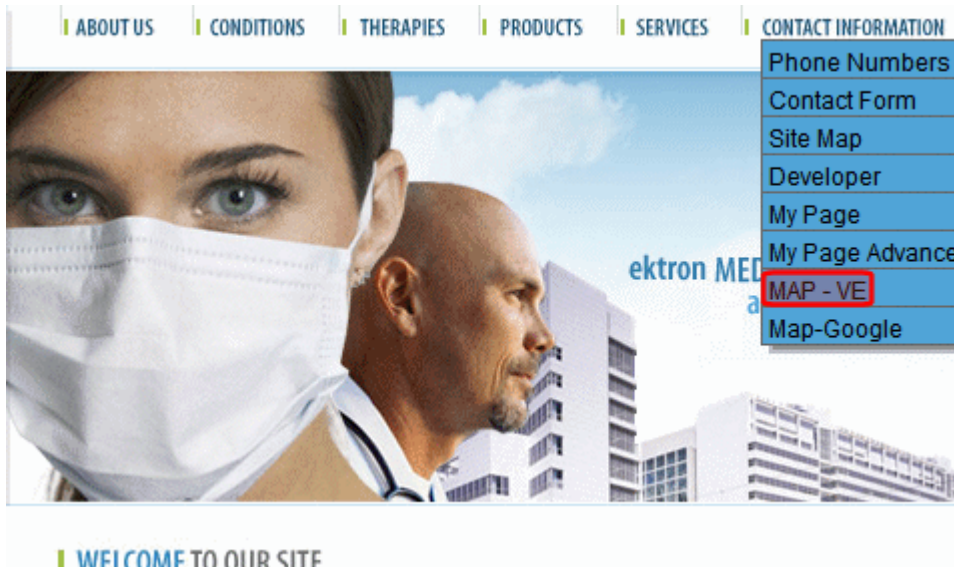
This section explains how site visitors can use the map's features, as well as how you can control the map's operation through the following sections.

- "Using Maps on the Sample Site" on page 191
- "Controlling the Visitor Experience with Maps" on page 200
- "Map Server Control Properties" on page 204

Using Maps on the Sample Site

You can view maps on **Ektron CMS400.NET**'s sample site. To do so, open the home page (<http://your server address/siteroot/default.aspx>). Then, click **Contact Information > Map VE** or **Contact Information > Map Google**.

NOTE If you want to work with these maps in VisualStudio.NET, the Virtual Earth map is on the [map.aspx](#) page. The Google map is on [mapgoogle.aspx](#).



NOTE Ektron CMS400.NET provides a Google map key that you can use for testing on localhost. But, if you want to use Google map with your production server, you must install a free license key. See "Obtaining a License to Use Google Maps" on page 192.

Obtaining a License to Use Google Maps

Before your production server can use Google's map feature, follow these steps to obtain and install a free license key.

1. Go to <http://www.google.com/apis/maps>.
2. Click **Sign up for a Google Maps API key**.
3. Read the terms and conditions. At the bottom of the page, check the box next to **I have read and agree with the terms and conditions**.
4. Enter your URL in the **My web site URL:** field. For example, `http://www.example.com/maps`.

IMPORTANT! A single Maps API key is valid for a single "directory" on your web server. If you sign up for the URL `http://www.example.com/maps`, the key you get is good for all URLs in the `http://www.example.com/maps/` directory.

5. Click the **Generate API Key** button.

the limitations and conditions described below. The API is limited to allowing You to display map images only, and does not provide You with the ability to access the underlying map data, any services provided by Google in connection with its maps service (such as local search or directions), or any other Google service.

☒ I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

6. You are asked to sign in to Google. If you don't have an account, create one.
7. A new page appears with your license key at the top. Copy the key.

Google Maps API

Thank you for signing up for a Google Maps API ke

Your key is:

```
ABQIAAAArjzjmEV4VOKv73AH3sZAhS2NFnaPLGeIgVqM-YohyxEGD_kPxTwsMJct30
```

This key is good for all URLs in this directory:

```
http://www.myelektronsite.com/maps/
```

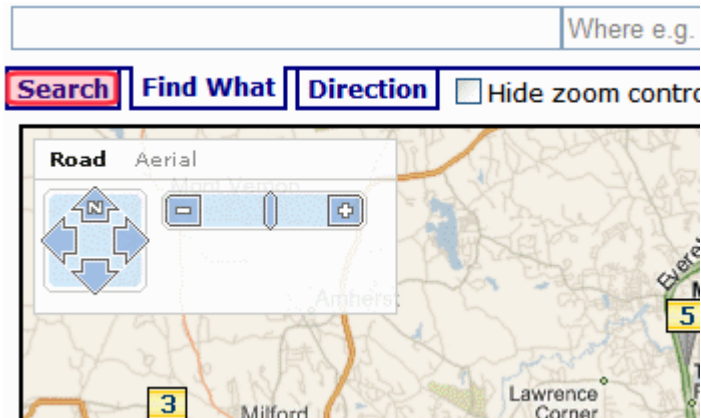
8. Without closing that page, open the `web site root/web.config` file.
 9. Go to the line that begins `<add key="GoogleMap"`.
 10. Within that line, replace the localhost key (shown below) with the one copied in Step 7
- ```
key=ABQIAAAAKS0ea074jIASulUo1JlxPRQAPC5zwhKxdscXMmLMOnW7z2LmSRQrVhf
XVp1hq2cv_ZsOvqP4GQ69eg" />
```

## Changing the Map's Zoom Level

If the map has a zoom control, the site visitor can use it to zoom in and out. As the map's boundaries change, flagged locations may appear or disappear. This is because the map only shows content whose location lies within the map's boundaries. See Also: ["Controlling the Visitor Experience with Maps" on page 200](#)

## Using the Search Tab to Recenter the Map

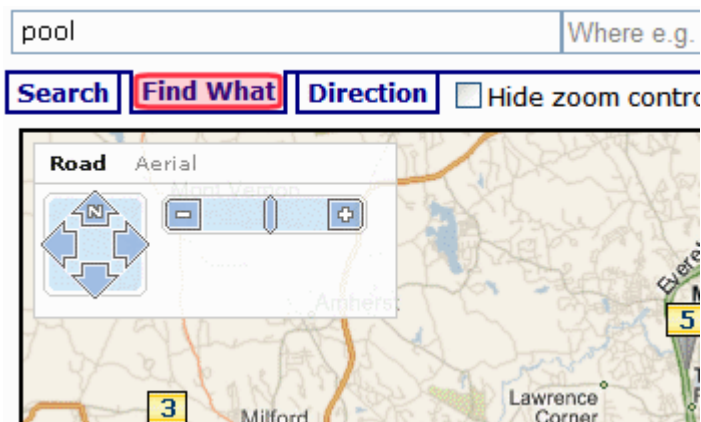
The **Search** tab lets a site visitor recenter the map on a location. He can enter any combination of street address, city, state and zip code.



When he does and presses **Search**, only content within that geographic area which satisfies other search criteria appears.

## Using the Find What Tab to Find Locations with a Search Term

The site visitor can use the **Find What** tab to find only locations that include a search term. For example, if a map flags hotels, the visitor can click the **Find What** tab then insert **pool** in the text box above to view only hotels with a pool.



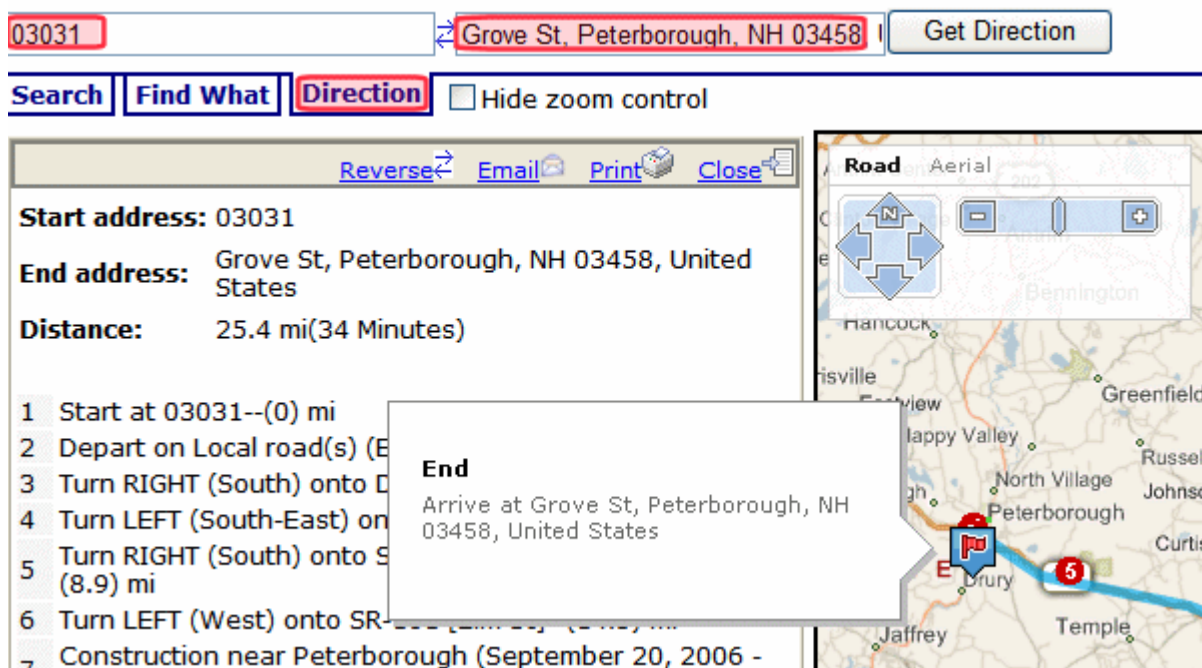


The **Find What** tab uses the same logic used in the Web Search to find content on your site. This is described in the **Ektron CMS400.NET** Administrator Manual section "Searching Your Web Site."

## Using the Directions Tab

The site visitor can use the **Directions** tab to get directions between any two locations. Simply enter any combination of street address, city, state, and zip code into both text boxes above the tabs (circled below).

When you do this and press **Get Directions**, the map screen displays text directions on the left, and a map of the directions on the right.



## The Display on Map Button

If you use **Ektron CMS400.NET**'s Taxonomy feature to classify mapped content, a site visitor can click the **Display on Map** button to restrict the map's locations to content in your categories.

**NOTE** The popup window also prompts the site visitor select one or a range of dates, and only retrieves content to which one of the selected dates is assigned. See "DateSearch" on page 207.

Display on Map

Clear All X

Date:

☐ Single Day
 ☒ Range

Start Date:

Thursday, May 03, 2007

End Date:

Friday, June 22, 2007

Categories:

☒ Restaurant  
☐ Business  
☐ Transportation

A good example is presented in **Ektron CMS400.NET**'s sample site. Here, the Taxonomy "Map Category" has the following categories.

View all categories of taxonomy "MapCategory"

View In: English (U.S.) Add: -select

MapCategory (17)
 

- Sports (0)
- Banks (0)
- Pharmacies (0)
- Travel (0)
- Restaurant (14)
  - American (2)
  - Indian (0)
  - Chinese (1)
  - Pizza (11)
  - Italian (0)
- Lodging (0)
- Parks and Recreation Areas (0)
- Community Services (0)
- Buisness (1)
- Transportation (2)

Notice that the category **Restaurant** has five subcategories.

When a site visitor is using the map, he can click the **Display on Map** button, see your content's categories, and select those of interest. When he does, the map updates to show content in selected categories only.

**NOTE** While an OR logical relationship among selected categories is the most intuitive and common, you can alternatively set up an AND or NOT logical relationship among selected categories. See "SelectedCategoryLogicalType" on page 211.

To learn about assigning taxonomy categories to content, see the **Ektron CMS400.NET** Administrator Manual chapter "Taxonomy Feature."

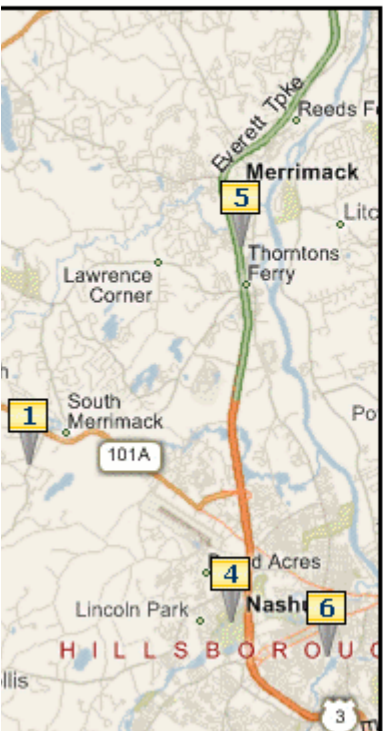
### Displaying Icons on the Category Popup Screen

See "CustomIcon" on page 206.

### Text Box

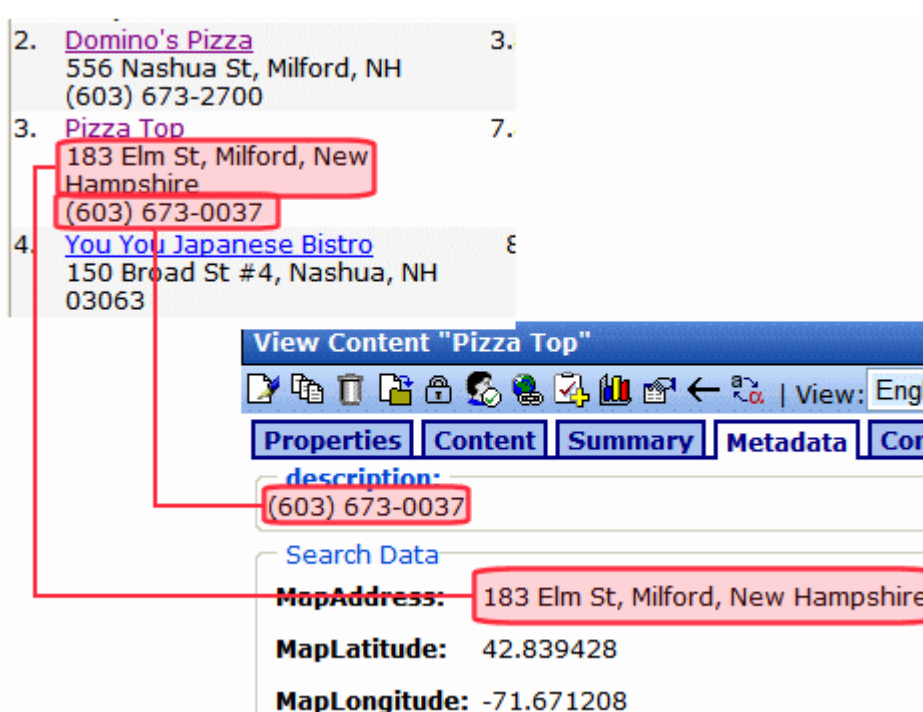


The text box appears to the right of the map if at least one location is flagged.

☐ Hide zoom control



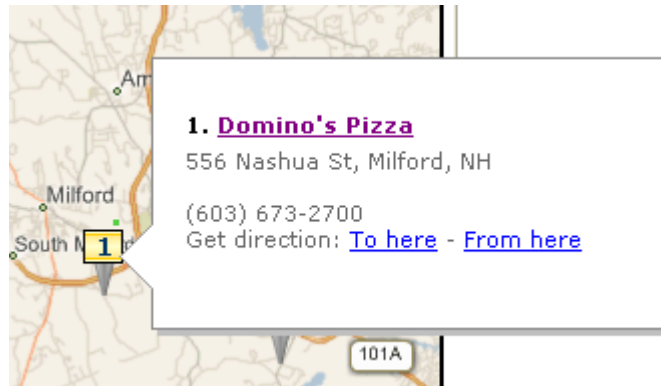
| No. | Title                                                                                                                           | Distance | Map | Direction |
|-----|---------------------------------------------------------------------------------------------------------------------------------|----------|-----|-----------|
| 1.  | <a href="#">Amherst House of Pizza</a><br>131 State Route 101A # 6,<br>Amherst, NH<br>(603) 886-5543                            | 1.45 Mi  |     |           |
| 2.  | <a href="#">Domino's Pizza</a><br>556 Nashua St, Milford, NH<br>(603) 673-2700                                                  | 3.83 Mi  |     |           |
| 3.  | <a href="#">Pizza Top</a><br>183 Elm St, Milford, NH<br>(603) 673-0037                                                          | 7.89 Mi  |     |           |
| 4.  | <a href="#">You You Japanese Bistro</a><br>150 Broad St #4, Nashua, NH<br>03063<br>Japanese, Korean, EurAsian Bistro            | 8.5 Mi   |     |           |
| 5.  | <a href="#">Budkley's Steak House</a><br>438 Daniel Webster Highway,<br>Merrimack, NH 03054<br>Steak wood grilled to perfection | 9.42 Mi  |     |           |
| 6.  | <a href="#">Surf Restaurants</a><br>207 Main Street, Nashua, NH<br>03060<br>Serious Seafood                                     | 11.29 Mi |     |           |

The text box's columns are described below.

| Column Header                                                                                  | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| No.                                                                                            | Map item's number. The lowest numbered item is closest to the map's center.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Title                                                                                          | <p>The content item that corresponds to the numbered map location. The site visitor can click the title to proceed to that content on your Web site.</p> <p>The text below the title is the Metadata <b>Map Address</b> field, followed by the Metadata <b>Description</b> field (see illustration below).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|                                                                                                |  <p>The screenshot illustrates the relationship between a map item and its metadata. On the left, a list of items is shown: 2. Domino's Pizza, 3. Pizza Top, and 4. You You Japanese Bistro. Red boxes highlight the address '(603) 673-0037' for Pizza Top and '183 Elm St, Milford, New Hampshire' for You You Japanese Bistro. On the right, the 'View Content "Pizza Top"' window is shown. It has tabs for Properties, Content, Summary, Metadata, and Content. The 'Content' tab is active, showing the 'description:' field with the value '(603) 673-0037'. Below this is the 'Search Data' section, which includes the 'MapAddress:' field with the value '183 Elm St, Milford, New Hampshire', and the 'MapLatitude:' (42.839428) and 'MapLongitude:' (-71.671208) fields. Red lines connect the highlighted address in the list to the 'MapAddress' field in the detailed view.</p> |
| Distance                                                                                       | Distance from the map's center to this content location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Map         | Click icon to recenter map on this content's location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Directions  | Click icon to select the <b>Directions</b> tab and paste this content location's address into the end location box. The site visitor could then enter a starting location to get directions to this location.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

## Bubble Text

When a site visitor hovers the cursor over a numbered map location, a text “bubble” appears.



The bubble shows the following information.

- Title - The content item that corresponds to the numbered map location. The site visitor can click the title to proceed to that content on your Web site.
- Summary - Whatever is entered into the content item's **Summary** tab.
- Metadata description - Whatever is entered into the content item's **Metadata Description** field.

 A screenshot of a web application window titled "Edit Content in Folder 'Map'". The window has a toolbar with icons for save, undo, redo, close, and help. Below the toolbar, there is a "Title:" label followed by a text box containing "Pizza Top". Below the title, there are five tabs: "Content", "Summary", "Metadata", "Schedule", and "Comments". The "Summary" tab is selected. Under the "Summary" tab, there is a label "description:" in a red box, followed by a text box containing "(603) 673-0037". At the bottom, there is a "Default" button and a text box showing "current character count: 14 (500 max)".

- Get Directions - Click **To Here** or **From Here** to select the **Directions** tab and paste this location's address into the To or From location box. The site visitor would then supply the missing (starting or ending) location to get directions between the locations.

## Controlling the Visitor Experience with Maps

**TIP!** Within Visual Studio.Net, you cannot see the map in design mode. But, you can right click the mouse and select **View in Browser** to see the effect of changing properties in a browser.

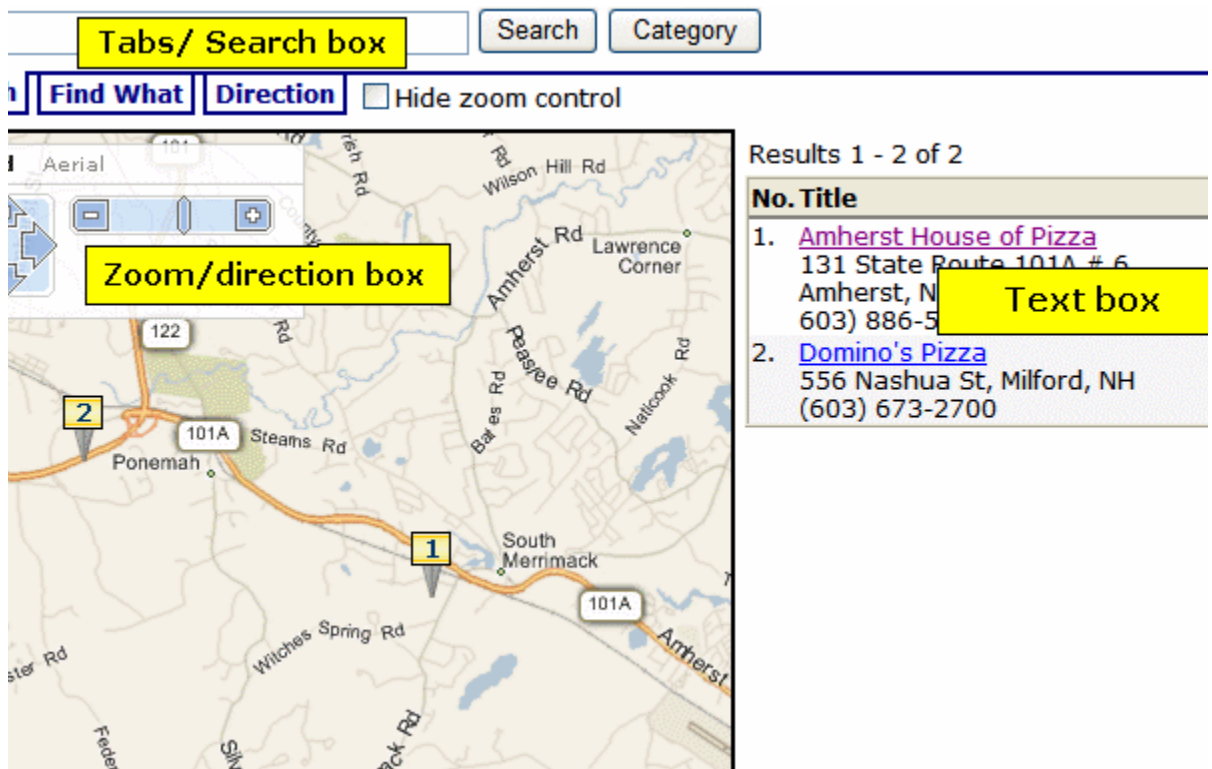
This section explains how to customize the following elements of the map display.

- "Displaying/Suppressing Map Elements" on page 200
- "Determining a Map's Initial Boundaries" on page 201
- "Determining which Content is Found by a Map" on page 201

For information on customizing the display using the Ektron Markup Language, see "map.ekml" on page 787

### Displaying/Suppressing Map Elements

You can use Map server control properties to display or suppress the following map elements.



| Screen Area                             | Use this Map server control property to display or suppress                                                 |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------------|
| Tabs/Search box                         | GeoControl                                                                                                  |
| Text box                                | DisplayTextResult                                                                                           |
| Zoom/direction                          | ZoomControl<br><br><u>Note: For Virtual Earth maps, this property also controls the Map type selection.</u> |
| Map type (road, satellite, combination) | TypeControl<br><br><u>Note: Only affects Google maps.</u>                                                   |

## Determining a Map's Initial Boundaries

Because the map only displays content whose address lies within the map's boundaries, focus the initial display on your businesses/locations. Map server control properties let you specify a beginning address (or longitude/latitude) and a starting zoom level. All content with address data within that area is flagged on the map.

If your locations are too spread out to appear on a single map, create several regional maps. Each map server control must appear on a separate web form.

---

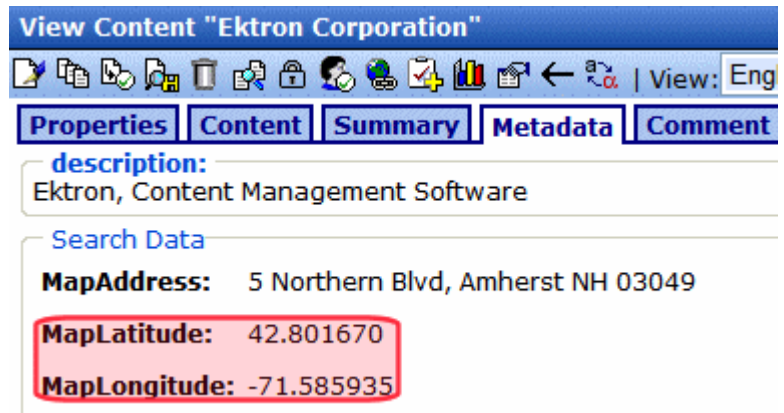
**WARNING!** You cannot place more than one map server control on a Web form.

---

## Determining which Content is Found by a Map

### *Preparing CMS Content to Appear on a Map*

Whether you use Google or Virtual Earth maps, content must have latitude and longitude values in order to appear on a map.



Google maps provides a service that takes the address of content and returns its latitude and longitude.

**NOTE** You don't need to use Google's automatic retrieval of latitude and longitude. Instead, you can enter the values manually. To do so, open the content item, go to its metadata, and enter the latitude and longitude values under **Search Data**.

To automatically obtain latitude and longitude information for any content item, follow these steps.

**IMPORTANT!** The following procedure assumes you are using Ektron CMS400.NET's sample site. If you are using the Min site, you must create searchable metadata definitions for **Map Address**, **Map Latitude** and **Map Longitude**. When defining **Map Latitude** and **Map Longitude**, set their **Style** to **Double**. If you are using dates with metadata, set **MapDate**'s style to **Date**.



**Edit Metadata definition "MapLatitude"**

**Name:** MapLatitude  
**ID:** 110  
**Type:** Searchable Property  
**Editable:** ☒  
**Display in Ektron Explorer** ☐

**Searchable Property**

**Publicly viewable:** ☐  
**Style:** Double  
**Default:**

Text  
 Number (generic)  
 Byte  
 Double  
 Float

1. Obtain a Google maps license key. See ["Obtaining a License to Use Google Maps"](#) on page 192

**NOTE** Step 2 requires the installation of the Developer SDK. For more information on this, see ["Installing the Developer SDK"](#) on page 806.

2. Move the GoogleGeocoder.dll file from C:\Program Files\Ektron\CMS400SDK\Samples\GoogleGeoCoder\obj\Release to C:\Program Files\Ektron\Plugins\Extensions.

**NOTE** Only move this file if you will use the mapping feature. If you will not, moving this file to the Extensions folder might slow down your server.

3. Create the content if necessary. On the content's **Metadata** tab, under **Search Data**, enter a **MapAddress** (that is, any combination of street, city, state, and zip code).

If you enter only a zip code, the latitude and longitude are set to that post office.

4. Publish the content. This action creates the Web Service call to Google maps, which retrieves the latitude and longitude for each address.

If Google maps cannot find a latitude and longitude (usually due to insufficient or conflicting information), it writes failure information to your server. You can view this under Windows Event Viewer > EktronLog. Any event's properties explain why the retrieval of latitude and longitude failed.

### ***Restricting Content for a Particular Map***

You may want a map to show a subset of all content with a latitude and longitude. For example, your business includes restaurants and bakeries, and you want a map to show only bakeries.

To accomplish this, place content for restaurants in one folder, and bakeries in another. Then, in the map server control that shows bakeries only, at the **FolderID** property, identify the bakeries folder.

If you want another map to show both restaurants and bakeries, create the restaurant and bakery folders under a parent folder. Then, in that map server control's **FolderID** property, identify the parent folder, and set the **Recursive** property to **true**.

## Map Server Control Properties

---

**WARNING!** You cannot place more than one map server control on a form.

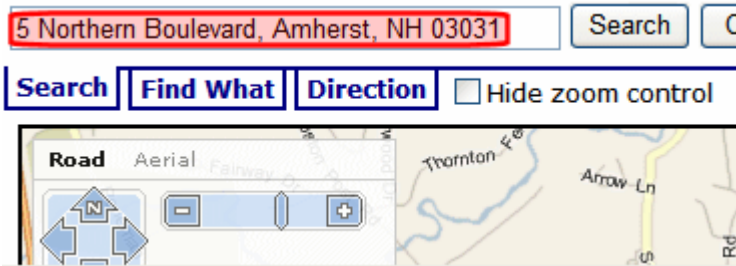
---

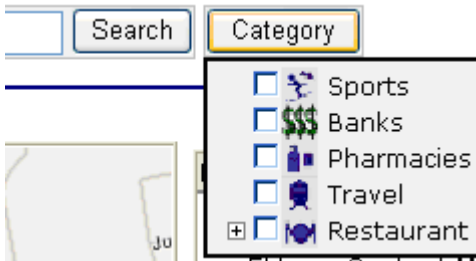
**Tip!** Within Visual Studio.Net, you cannot see the map in design mode. However, you can right mouse click the mouse and select **View in Browser** to see the effect of changing the properties.

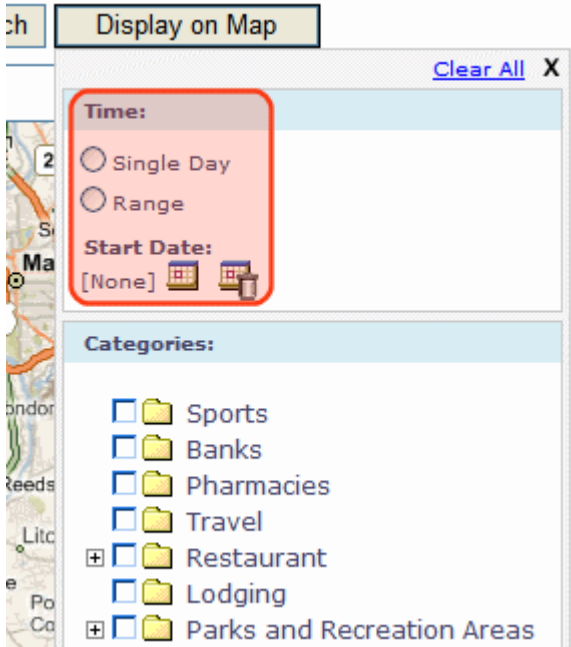
The following table explains the Map server control properties.

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

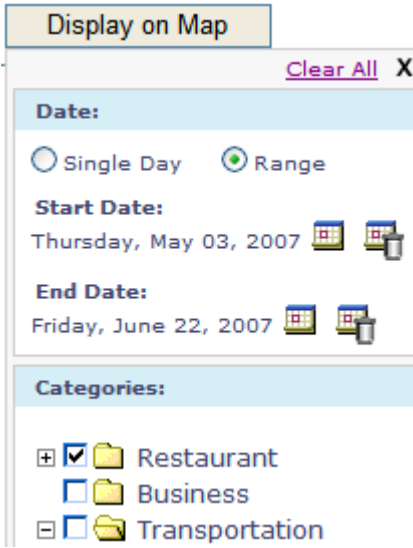
**NOTE** Do not use a percentage value to set the native .NET properties *Height* and *width*. Percentages do not work with these properties.

| Property      | Value                                                                                                                                                                                                                                                                                                                                                                                                                    | Data Type |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Address       | <p>To set a default map center, enter its address here. If you only enter a zip code, the map centers on its post office.</p> <p>The address appears in the <b>Search</b> field when the map first displays.</p>  <p>If you enter an address, the latitude and longitude properties are ignored.</p>                                  | String    |
| Authenticated | <p>Indicates if you are logged in to the CMS Explorer and can use it to browse to content and a Taxonomy. If you are not logged it but want to, change the value to <b>true</b>.</p> <p>See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a></p>                                                                                                                              | Boolean   |
| ContentId     | <p>Use this field in conjunction with the <b>EnableSearchResult</b> field to limit the map to a single content item. Here, you identify the content item to be mapped. See Also: <a href="#">"EnableSearchResult" on page 208</a></p> <p>As explained in <a href="#">"Preparing CMS Content to Appear on a Map" on page 201</a>, content <i>must</i> have latitude and longitude values in order to appear on a map.</p> | Integer   |

| Property   | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Data Type |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| CustomIcon | <p>Use this field if you want the Category popup box to display an icon to the left of each category, as shown below.</p>  <p><b>Prerequisite</b></p> <p>To use this value, open the <code>webroot/Workarea/images/application/maps/tree</code> folder. In that folder, create a new folder whose name is the same as the Taxonomy category assigned to the Map server control at the MapCategory property. In the sample site, this Taxonomy's name is MapCategory.</p> <p>Then, place the icons in that folder by category title name with a .png extension. Use an underscore ( <code>_</code> ) to separate taxonomy levels. For example, the image for the category Restaurant &gt; American must be named <code>Restaurant_American.png</code>.</p> | Boolean   |

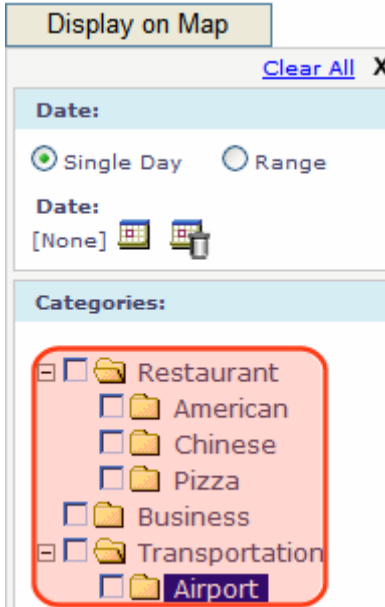
| Property          | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Data Type   |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| DateSearch        | <p>If you want to assign a date to content so that site visitors can select by date, enter <b>true</b>. This feature is helpful for date-related content, such as concerts, meetings, or sporting events. The site visitor can filter what appears on the map by date.</p> <p>If you set this property to true, assign a date to date-related content using the <b>MapDate</b> standard metadata field. Then, publish the content.</p> <p>When a site visitor clicks a map's <b>Display on Map</b> button, the popup screen includes date criteria, as shown below. See Also: <a href="#">"The Display on Map Button" on page 195</a></p>  <p>The site visitor can select map items by a single date or a range of dates. If he does, only content to which one of the selected dates is assigned appears.</p> | Boolean     |
| DisplayTextResult | <p>If you want the site visitor to see a box of information about each map item to the right of a map, enter <b>true</b>. To suppress the text box, enter <b>false</b>.</p> <p>See Also: <a href="#">"Displaying/Suppressing Map Elements" on page 200</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Boolean     |
| DistanceUnit      | Enter the units of distance on the map. Choices are miles and kilometers. The default value is miles.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | enumeration |


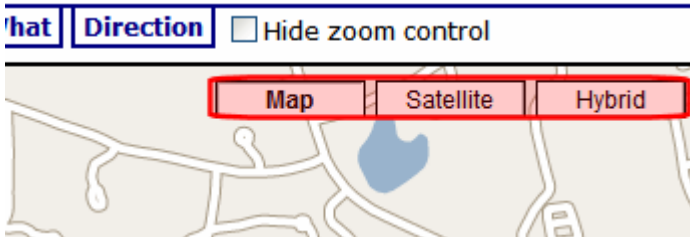
| Property           | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Data Type |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| EnableSearchResult | <p>Use to determine if the Map server control accesses the CMS search to return results. <b>True</b> is the default value.</p> <p>You would change it to <b>False</b> if you want to display a map with only one content item, that is, a single location. Specify this item at the <b>ContentID</b> property. In this case, the map does not find other CMS content whose longitude and latitude are within the map's boundaries.</p> <p>Site visitors viewing the map can still use the <b>Search</b> and <b>Directions</b> tabs to get directions to the location. See Also: <a href="#">"Using the Search Tab to Recenter the Map" on page 194</a>; <a href="#">"Using the Directions Tab" on page 195</a></p> <p>For example, your site features homes for sale, and you want a map to show a featured home of the week. To do so, set this property to <b>False</b>, and enter the content block that describes that home in the <b>ContentID</b> field. See Also: <a href="#">"ContentId" on page 205</a></p> | Boolean   |
| FolderID           | Enter the ID number of the folder whose content is searched on this map. If the <code>recursive</code> property is true, folders below this folder are also searched.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Integer   |
| GeoControl         | <p>If you want the site visitor to see a search box with tabs above a map, enter <b>true</b>. To suppress the text box, enter <b>false</b>.</p> <p>See Also: <a href="#">"Displaying/Suppressing Map Elements" on page 200</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | Boolean   |
| Hide               | <p>Used to hide the map server control in design time and run time.</p> <p><b>True</b> = Hide map server control</p> <p><b>False</b> = Show map server control</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | Boolean   |
| Language           | <p>Set a language for the Map server control. To allow the language to be set dynamically, enter zero (0).</p> <p>The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Integer   |
| Latitude           | To set a default map center via latitude and longitude (as opposed to an address), enter the latitude here.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Decimal   |
| Longitude          | To set a default map center via latitude and longitude (as opposed to an address), enter the longitude here.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Decimal   |

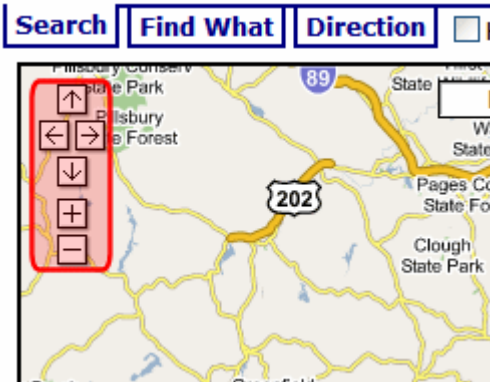
| Property    | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Data Type   |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| MapCategory | <p>Enter the ID number of the taxonomy whose categories appear when a site visitor clicks this map's <b>Display on Map</b> button.</p> <p>When a map first appears, all eligible content appears. If a site visitor clicks the <b>Display on Map</b> button, he can choose one or more categories and limit the map to items assigned to them. For example, the user could view restaurants only (as shown below).</p> <p>As another example, your map could display all campuses in your state's college system. A site visitor could use the <b>Display on Map</b> popup window to limit the map to community colleges.</p> <hr/> <p><a href="#">Note: While an OR logical relationship among selected categories is the most intuitive and common, you can alternatively set up an AND or NOT logical relationship among selected categories. See "SelectedCategoryLogicalType" on page 211.</a></p> <hr/> <p>To learn more about Taxonomy, see the <b>Ektron CMS400.NET</b> Administrator Manual chapter "Taxonomy."</p> | Integer     |
|             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |             |
| MapProvider | Select the service that provides the map, either Google or Virtual Earth.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Enumeration |
| MapStyle    | <p>Enter the map's display mode: Road, Satellite or Hybrid.</p> <hr/> <p><a href="#">Note: This setting only affects Virtual Earth maps.</a></p> <hr/>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Enumeration |

| Property        | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Data Type |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|-------|----------|----|------------------------------------------------------------------------------------------------|-------|----|-----------------------------------------------------------------------------------------|-------|----|----------------------------------------------------------------------------------|-------|---------|
| Markup Language | <p>Enter the template markup file that is used to control the display of the map page. For example, mymapmarkup.ekml.</p> <p>If you enter no EkML file, the default one at Workarea\Template\map\map.ekml is used.</p> <p>If the *.ekml file is located in the \workarea\template\map folder, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \workarea\customfiles\markup\mymapmarkup.ekml.</p> <p>See Also: "Ektron Markup Language" on page 754 and "map.ekml" on page 787</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                  | String    |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| MinZoomLevel    | <p>If you want to set a map zoom level below which map locations will not appear, enter that value.</p> <p>The default value is 4. Possible values are between 1 (most detailed) and 19.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |           |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| PageSize        | <p>Enter the number of locations that can appear on one page of the text box after a search is executed. See Also: "Displaying/ Suppressing Map Elements" on page 200</p> <p>If more than this number of locations are available, the site visitor sees <b>[First] [Previous] [Next] [Last]</b> at the bottom of the page. He can use the navigation aids to view additional locations.</p> <table border="1"> <thead> <tr> <th>No.</th> <th>Title</th> <th>Distance</th> </tr> </thead> <tbody> <tr> <td>1.</td> <td><a href="#">Uno Restaurant &amp; Bar</a><br/>1875 S Willow St, Manchester, NH<br/>(603) 647-8667</td> <td>18.99</td> </tr> <tr> <td>2.</td> <td><a href="#">Alley Cat Pizzeria</a><br/>486 Chestnut St, Manchester, NH<br/>(603) 669-4533</td> <td>22.61</td> </tr> <tr> <td>3.</td> <td><a href="#">Xpress Pizza</a><br/>108 Webster St, Manchester, NH<br/>(603) 641-3600</td> <td>23.59</td> </tr> </tbody> </table> <p><b>[First] [Previous] [Next] [Last]</b></p> | No.       | Title | Distance | 1. | <a href="#">Uno Restaurant &amp; Bar</a><br>1875 S Willow St, Manchester, NH<br>(603) 647-8667 | 18.99 | 2. | <a href="#">Alley Cat Pizzeria</a><br>486 Chestnut St, Manchester, NH<br>(603) 669-4533 | 22.61 | 3. | <a href="#">Xpress Pizza</a><br>108 Webster St, Manchester, NH<br>(603) 641-3600 | 23.59 | Integer |
| No.             | Title                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Distance  |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| 1.              | <a href="#">Uno Restaurant &amp; Bar</a><br>1875 S Willow St, Manchester, NH<br>(603) 647-8667                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | 18.99     |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| 2.              | <a href="#">Alley Cat Pizzeria</a><br>486 Chestnut St, Manchester, NH<br>(603) 669-4533                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 22.61     |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| 3.              | <a href="#">Xpress Pizza</a><br>108 Webster St, Manchester, NH<br>(603) 641-3600                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | 23.59     |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |
| Recursive       | <p>In the <b>FolderID</b> property, you specify a folder whose content is searched on this map. To extend the search to all folders below this folder, set this property to <b>true</b>.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Boolean   |       |          |    |                                                                                                |       |    |                                                                                         |       |    |                                                                                  |       |         |



| Property                    | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Data Type                  |
|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------|
| SelectedCategoryLogicalType | <p>Use this property to determine the logical relationship among several categories on the Display on Map tab. (See <a href="#">"The Display on Map Button" on page 195</a>). There are three choices: OR, AND, and NOT.</p>  <p>By default, an OR relationship exists among the site visitor's selections. So, for example, if all three Restaurant categories are checked, then any restaurant that is defined as American, Chinese or Pizza appears on the map.</p> <p>If you change this property's value to AND, only content to which <i>all</i> selected categories apply appear on the map. In the above example, only restaurants defined as American <i>and</i> Chinese <i>and</i> Pizza appear on the map.</p> <p>Alternatively, you can set the property's value to NOT. In this case, only content to which the selected categories are not applied appear on the map. To continue the above example, if the site visitor selects <b>Chinese</b>, only restaurants that are not assigned the Chinese category appear on the map.</p> | Enumeration (Or, And, Not) |
| StartZoomLevel              | Enter the zoom level at which the map initially appears. See <a href="#">Also: "Determining a Map's Initial Boundaries" on page 201</a>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Integer                    |

| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Data Type |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| SuppressWrapperTags | <p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags</p> <p><b>False</b> - Allow wrap tags</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Boolean   |
| TypeControl         | <p>If this map uses Virtual Earth maps, this property enables or disables the zoom/direction/type control (highlighted below). It lets a site visitor zoom the map in and out, move the center in any direction, and change the display style (Road or Aerial).</p>  <p>If this map uses Google Earth maps, this property enables or disables the type control (highlighted below). It lets the site visitor change the display style (Map, Satellite, or Hybrid). Use the ZoomControl property to display or suppress Google Earth's zoom and direction controls.</p>  | Boolean   |
| WrapTag             | <p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p>                                                                                                                                                                                                                                                                                                                                                        | String    |

| Property    | Value                                                                                                                                                                                                                                                                                               | Data Type |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| ZoomControl | <p>Use this property to display or suppress the Google map zoom control (highlighted below).</p>  <p>For Virtual Earth maps, the display of the zoom control is determined by the <b>TypeControl</b> property.</p> |           |

# Membership Server Control

The Membership server control creates a form on a Web page that allows a user to create or update a profile of themselves in your membership list. The control creates the form based on the standard Ektron user properties and any custom user properties that exist. It also provides a way to have new membership users read and accept the terms and conditions for using Discussion Boards (Forums) when needed.

Below is an example of the Membership server control when viewed on a Web page.

The screenshot shows a web form for user registration, divided into three main sections with colored borders and labels:

- Standard User Properties (Red border):** This section contains fields for \*First Name, \*Last Name, \*Password, \*Confirm Pwd, \*E-Mail Address, Display Name, and User Language (set to English (U.S.)).
- Custom User Properties (Blue border):** This section includes a Subscriptions checkbox for Wellness Articles (with a note: Notification will send in user language), a zip field, a Private Profile checkbox, and a \*Region dropdown menu (set to Select Region).
- Terms and Conditions (Green border):** This section contains text stating: "In order to proceed, you must agree with the following rules: Use of this Site constitutes agreement with the following terms and conditions: The Forum administers this Site. Unless expressly stated otherwise, the findings interpretations and conclusions expressed in the materials in this Site are those of the various authors of the work and are not necessarily those of The Forum's staff or Board..." Below this text is a checkbox labeled "I have read and agree to abide by the forum rules."

At the bottom of the form are two buttons: Register and Reset.

When you create custom user properties, they appear for both membership users and **Ektron CMS400.NET** users. If you create a property such as Employee ID

number, the property appears when a membership user views this form on a web page. The Membership server control allows you to decide if want the custom user properties to appear when the form is viewed.

It is recommended that you review the following documentation when using the Membership server control.


- The Administration Manual section “Managing Users & User Groups”
- The Administration Manual section “Custom User Properties”
- The Administration Manual section “Web Alert Feature”
- The Administration Manual section “Discussion Boards” > “Implementing Discussion Boards” > “Working with Discussion Boards and Categories” > “Terms and Conditions”

Understanding these topics along with using this server control extends the most functionality to your membership users.

## Membership Server Control Properties

The following table explains the properties of the Membership server control.

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

| Property      | Value                                                                                                                                                                                                                                                                                                                                                                                                                                              | Data Type |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Authenticated | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                                                                                                                                                                          | String    |
| BoardID       | <p>The ID of the discussion board for which to show the Terms and Conditions. See Also: <a href="#">"ShowTerms" on page 221</a>.</p> <p>If you don't know the ID, click the Ellipses button (  ), then sign in, browse to and select the discussion board. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a></p> |           |

| Property          | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Data Type |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| DisplayMode       | <p>The type of membership form displayed. Choose from the following.</p> <ul style="list-style-type: none"> <li>• <b>UserRegistration</b> - Lets a site visitor register as a membership user.</li> <li>• <b>ResetPassword</b> - Lets a membership user reset his password.</li> <li>• <b>UnsubscribeSecured</b> - Lets a membership user unsubscribe. He needs to enter his username and password.</li> <li>• <b>UnsubscribeUnsecured</b> - Lets a membership user unsubscribe by entering his membership username only.</li> <li>• <b>AccountActivate</b> - Let a site visitor activate his membership account by entering the ID number in the account verification email.</li> </ul> <p>For more information, see <a href="#">"DisplayMode" on page 219</a>.</p> | String    |
| Hide              | <p>Used to hide login button in design time and run time.</p> <p><b>True</b> = Hide login button</p> <p><b>False</b> = Show login button</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Boolean   |
| Language          | <p>Set a language for the server control. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Integer   |
| RedirectFailedURL | <p>The URL where a membership user is sent if the registration fails.</p> <ul style="list-style-type: none"> <li>• If the page resides in the same folder as the registration page, enter the name of the page. For example, <b>RegFailed.aspx</b>.</li> <li>• If the redirect page is located in a subfolder, add the folder path. For example, <b>members\RegFailed.aspx</b>.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                           | String    |
| RedirectSucessURL | <p>The URL where a membership user is sent when the registration succeeds.</p> <ul style="list-style-type: none"> <li>• If the page resides in the same folder as the registration page, enter the name of the page. For example, <b>RegSucceed.aspx</b>.</li> <li>• If the redirect page is located in a subfolder, add the folder path. For example, <b>members\RegSucceed.aspx</b>.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                    | String    |

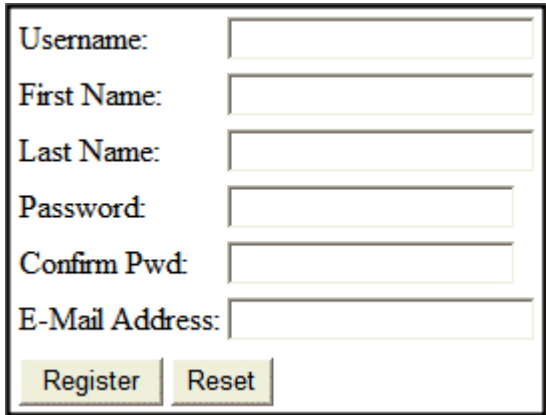

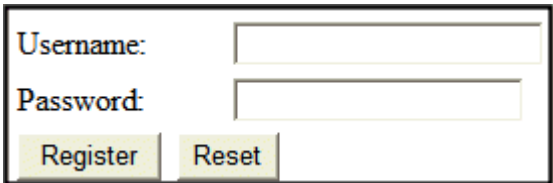

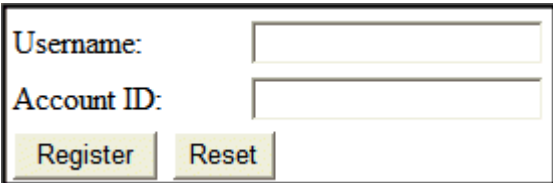
| Property           | Value                                                                                                                                                                                                                                                                                                                                                                     | Data Type |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| RegisterButtonImg  | <p>Lets you add an image in place of the text on the register button. Enter a path to the image. For example:</p> <pre>http://www.example.com/buttons/registerbutton.gif</pre> <p>If the image is located in the site root, you do not need to enter the full path. Only enter the subfolder path and image name. For example:</p> <pre>/buttons/registerbutton.gif</pre> | String    |
| RegisterButtonText | Text that appears on the Register button. The default is <b>Register</b> . If you use a register button image, you do not see this text.                                                                                                                                                                                                                                  | String    |
| ResetButtonImg     | <p>Lets you add an image in place of the text on the reset button. Enter a path to the image. For example:</p> <pre>http://www.example.com/buttons/resetbutton.gif</pre> <p>If the image is located in the site root, you do not need to enter the full path. Only enter the subfolder path and image name. For example:</p> <pre>/buttons/resetbutton.gif</pre>          | String    |
| ResetButtonText    | Text that appears on the reset button. The default is <b>Reset</b> . If you use a reset button image, you do not see this text.                                                                                                                                                                                                                                           | String    |
| SaveButtonImg      | <p>Lets you add an image in place of the text on the save button. Enter a path to the image. For example:</p> <pre>http://www.example.com/buttons/savebutton.gif</pre> <p>If the image is located in the site root, you do not need to enter the full path. Only enter the subfolder path and image name. For example:</p> <pre>/buttons/savebutton.gif</pre>             | String    |
| SaveButtonText     | Text that appears on the save button. The default is <b>Save</b> . If you use a save button image, you do not see this text.                                                                                                                                                                                                                                              | String    |
| ShowExtended       | <p>Decide if the Custom User Properties are available when using this control. The default setting is <b>True</b>.</p> <p><b>True</b> = Show Custom User Properties<br/> <b>False</b> = Hide Custom User Properties</p> <hr/> <p><b>Note:</b> For more information on Custom User Properties, see the Administration Manual section "Custom User Properties".</p> <hr/>   | String    |

| Property                 | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Data Type |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| ShowTerms                | <p>When set to True this property shows the Terms and Conditions as defined in the Workarea. In addition, users are asked to check a box stating they have read and agree to abide by the terms and conditions of the forum. Default is <b>False</b>.</p> <p>See Also: "<a href="#">ShowTerms</a>" on page 221 and the Administrator Manual section "Discussion Boards" &gt; "Implementing Discussion Boards" &gt; "Working with Discussion Boards and Categories" &gt; "Terms &amp; Conditions"</p> | Boolean   |
| SuppressWrapperTags      | <p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags.</p> <p><b>False</b> - Allow wrap tags.</p>                                                                                                                                                                                                                                                                                                                   | Boolean   |
| UserExistsMessage        | The message that appears when a membership user already exists. The default message is: <b>Username(email) already exists!</b>                                                                                                                                                                                                                                                                                                                                                                       | String    |
| UserSuccessMessage       | The message that appears when a membership user successfully registers. The default message is: <b>You have registered successfully.</b>                                                                                                                                                                                                                                                                                                                                                             | String    |
| UserUpdateSuccessMessage | The message that appears when a membership user successfully updates their information. The default message is: <b>You have successfully updated your information.</b>                                                                                                                                                                                                                                                                                                                               | String    |
| WrapTag                  | <p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p>                                                                                                                       | String    |



## DisplayMode

Lets a developer decide what type of membership form is added to the Web form. The following table describes settings for the DisplayMode property.

| Setting              | Description                                                                                                                                                                                                                                        | Example                                                                                                                                                                                                                                           |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UserRegistration     | Lets a site visitor register as a membership user.<br>This mode also allows membership users to update their information and preferences when they are logged in. See <a href="#">Also: "Updating a Membership User's Information" on page 220</a> |  <p>A registration form with fields for Username, First Name, Last Name, Password, Confirm Pwd, and E-Mail Address. It includes Register and Reset buttons.</p> |
| ResetPassword        | Lets a membership user reset his password.                                                                                                                                                                                                         |  <p>A form with a Username field and Register and Reset buttons.</p>                                                                                            |
| UnsubscribeSecured   | Lets a membership user unsubscribe. The membership user needs to enter his username and password to unsubscribe.                                                                                                                                   |  <p>A form with Username and Password fields and Register and Reset buttons.</p>                                                                              |
| UnsubscribeUnsecured | Lets a membership user unsubscribe by entering his membership username only.                                                                                                                                                                       |  <p>A form with a Username field and Register and Reset buttons.</p>                                                                                          |
| AccountActivate      | Let a site visitor activate his membership account by entering the ID number sent him in the account verification email.                                                                                                                           |  <p>A form with Username and Account ID fields and Register and Reset buttons.</p>                                                                            |


## Updating a Membership User's Information

Membership users can update their profile by returning to the membership registration page when they are logged in. The fields are filled in with the users information.

Users can also edit their forum preferences. This can not be done when a user is sign up for the first time.

### Editing a Membership User's Forum Properties

The table below describes the fields found on the Forum Properties screen.

| Field        | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Display Name | The name that is associated with the user when posting on the Forum.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Forum Editor | Choose the editor you want to use to create forum posts. The default is Javascript Editor. In addition you might have eWebEditPro as an option. In the instance of the Forum the Javascript Editor has more options.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Avatar       | <p>An image or icon to represent you in the Forum.</p> <p>Type in a web path to image. For example:</p> <p><a href="http://www.example.com/smileyface.gif">http://www.example.com/smileyface.gif</a></p>  <p>The screenshot shows a forum post interface. On the left, a user profile is displayed with the text 'JE', 'Joined: 3/7/2006', and 'Posts: 2'. Below this is a yellow smiley face icon with a red rectangular box around it. To the right of the profile, the post content begins with 'Posted: Tuesday, O' and 'This is a test repl'. Below the post content, the user's name 'John Edit' is shown, followed by 'Technical Editor' and 'Ektron, Inc.'. At the bottom of the post, there are buttons for 'ADD REPLY' and 'REPO', and a link for 'Back to top'.</p> |
| Page Size    | The number of entries per page.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

| Field           | Description                                                                                                                                                                                                                            |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Forum Signature | <p>Adds a signature to your posts in the forum. You do not see the signature in the editor. It is added after you click the <b>Post</b> button.</p>  |

## ShowTerms

Setting the ShowTerms property to True allows you to show the Terms and Conditions for the forum. In addition, users are asked to check a box stating they have read and agree to abide by the terms and conditions of the forum.

Terms and Conditions are added in the Workarea. *See Also:* The Administrator Manual section "Discussion Boards" > "Implementing Discussion Boards" > "Working with Discussion Boards and Categories" > "Terms & Conditions"

---

**NOTE** If there are no Terms and Conditions defined in the Workarea, Setting this property to true does nothing.

---

Setting this property to False hides the terms and conditions of the forum. If Terms and Conditions are defined in the Workarea and this property is set to False,

users will still have to read and accept the Terms and Conditions the first time they create a post or a reply.

\*First Name:

\*Last Name:

\*Password:

\*Confirm Pwd:

\*E-Mail Address:

Display Name:

User Language:

English (U.S.)

Subscriptions

☐ Wellness Articles  
(Notification will send in user language)

zip

Private Profile

☐

\*Region

Select Region

In order to proceed, you must agree with the following rules:  
Use of this Site constitutes agreement with the following terms and conditions:  
The Forum administers this Site. Unless expressly stated otherwise, the findings  
interpretations and conclusions expressed in the materials in this Site are those  
of the various authors of the work and are not necessarily those of The Forum's  
staff or Board...

☐ I have read and agree to abide by the forum rules.

Register

Reset

# Introduction to Menu Server Controls

**Ektron CMS400.NET** provides four menu server controls:

- ["Flexible Menu Server Control" on page 232](#)
- ["Smart Menu Server Control" on page 226](#)
- ["Menu Server Control" on page 246](#)
- ["DhtmlMenu Server Control" on page 254](#)

This section introduces concepts that affect all of them, and explains the differences between them.


## Prerequisites


Before you can use the server controls, you must create menus in the Workarea. To learn about creating menus, see "Working with Menus" in the **Ektron CMS400.NET** User Manual and "Menu Feature" in the **Ektron CMS400.NET** Administrator Manual.

## Contrasting Menu Server Controls

- ["Appearance" on page 223](#)
- ["Properties" on page 224](#)

### Appearance

|                            |                                                                                                                                               |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Flex and Smart Menu</b> | <div></div> <p>or</p> <p>(Other styles are available)</p> |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|

|                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>DHTML Menu</b> |  <p>(Styling is very customizable)</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Menu</b>       | <p><a href="#">Inside Ektron Medical</a></p> <ul style="list-style-type: none"> <li>○ <a href="#">About Us</a></li> <li>○ <a href="#">Conditions</a></li> <li>○ <a href="#">Therapies</a></li> <li>○ <a href="#">Products</a></li> <li>○ <a href="#">Services</a></li> <li>○ <a href="#">Contact Information</a></li> <li>○ <a href="#">secondary menu</a> <ul style="list-style-type: none"> <li>■ <a href="#">Where did you hear about Ektron Medical?</a></li> <li>■ <a href="#">Ektron</a></li> <li>■ <a href="#">Ektron CMS400.NET Logo - Reflection</a></li> </ul> </li> <li>○ <a href="#">Another submenu</a> <ul style="list-style-type: none"> <li>■ <a href="#">Sample Content Block</a></li> </ul> </li> </ul> |

## Properties

| Property                                                                        | Flex Menus | Smart Menu | DHTML Menu | Menu |
|---------------------------------------------------------------------------------|------------|------------|------------|------|
| 508 compliant                                                                   | ✓          | ✓          |            |      |
| Shutter menu<br>(When you click a menu, it opens directly below the menu title) | ✓          | ✓          |            |      |

| Property                                                                               | Flex Menus                                         | Smart Menu                                         | DHTML Menu                                | Menu                 |
|----------------------------------------------------------------------------------------|----------------------------------------------------|----------------------------------------------------|-------------------------------------------|----------------------|
| Secondary menu opens when cursor hovers over it                                        | ✓ (alternative: secondary menu opens when clicked) | ✓ (alternative: secondary menu opens when clicked) | ✓                                         |                      |
| Starts out collapsed; expands as site visitor selects secondary menu                   | ✓                                                  | ✓ (optional)                                       | ✓                                         |                      |
| All menu levels open when page displays                                                | ✓ (optional)                                       | ✓ (optional)                                       |                                           | ✓                    |
| Can display secondary menu to the right of the root menu, instead of directly below it | ✓ (optional)                                       | ✓ (controlled by style sheet)                      | ✓                                         |                      |
| How appearance is modified                                                             | Server control properties and style sheet          | Server control properties and style sheet          | Server control properties and style sheet | Xslt and style sheet |
| How behavior is modified                                                               | Xslt                                               | style sheet                                        | style sheet                               | Xslt and style sheet |
| Option to launch menu links in a new window                                            |                                                    | ✓                                                  |                                           |                      |

# Smart Menu Server Control

The Smart Menu server control displays a menu on a Web page.

Before you can use this server control, you must create a menu in the **Ektron CMS400.NET** Workarea. To learn about creating menus, see "Working with Menus" in the **Ektron CMS400.NET** User Manual and "Menu Feature" in the **Ektron CMS400.NET** Administrator Manual.

To learn about how the Smart Menus compare to other Ektron CMS400.NET menus, see ["Contrasting Menu Server Controls" on page 223](#).

## Smart Menu Server Control Properties

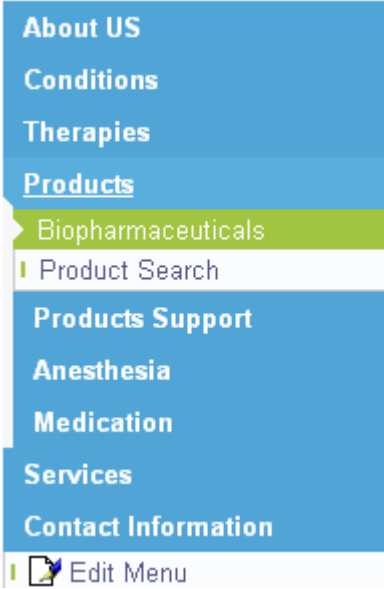
The Smart Menu server control properties are described in this table.

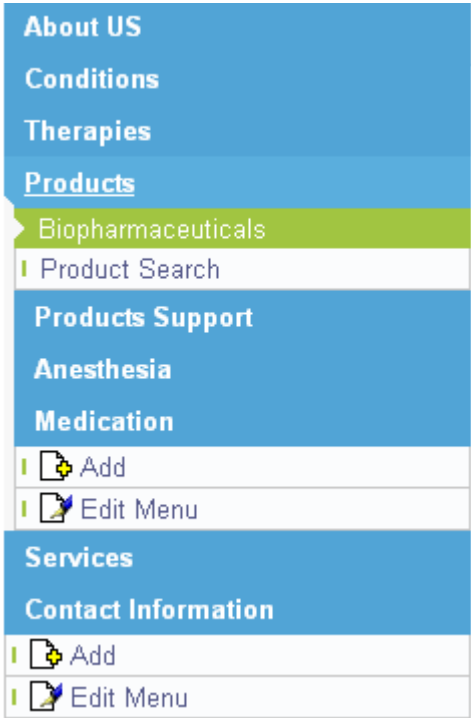
**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

| Property             | Value                                                                                                                                                                                                                                                                                                           | Data Type |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Authenticated        | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc.<br><i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                             | String    |
| AutoCollapseBranches | If you set to <b>true</b> , whenever a new submenu opens, all other submenus close.<br>If <b>false</b> , other submenus remain open when a new one opens.                                                                                                                                                       | Boolean   |
| CacheInterval        | Sets the amount of time the server control's data is cached. For example, if you want to cache the data for five minutes, set this property to 300. The default is zero.<br><i>See Also:</i> <a href="#">"Caching with Server Controls" on page 42</a> .                                                        | Integer   |
| DefaultMenuID        | The ID of a menu that appears where you insert this server control if no other menu is identified or available.<br>If you don't know the ID number of the menu, use the CMS Explorer to browse to it.<br><i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a> | Integer   |



| Property                | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Data Type |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| DynamicParameter        | Select <b>menu_id</b> . When you do, this server control uses the menu passed as a URL parameter.                                                                                                                                                                                                                                                                                                                                                                                                                             | String    |
| Enable508compliance     | If you set to <b>true</b> , the menu will comply with Section 508.                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Boolean   |
| EnableMouseOverPopup    | If you set to <b>true</b> , submenus appear as soon as the cursor moves over them.<br><br>If you set to <b>false</b> , submenus only appear if a site visitor clicks them or a keyboard equivalent.                                                                                                                                                                                                                                                                                                                           | Boolean   |
| EnableSmartOpen         | If you set to <b>true</b> , any submenu on the Smart Menu can be set to open automatically. To learn about how to do this, see the <b>Ektron CMS400.NET</b> Administrator Manual section "Menu Feature" > "Assigning Folders to a Menu."<br><br>If you set to <b>false</b> , even if all other conditions are present, submenus on a Smart Menu do not automatically open.<br><br>This property lets you design pages whose submenus do not normally open by default. Under some circumstances, such submenus look cluttered. | Boolean   |
| Hide                    | Used to hide a menu in design time and run time.<br><b>True</b> = Hide menu<br><b>False</b> = Show menu                                                                                                                                                                                                                                                                                                                                                                                                                       | Boolean   |
| Language                | Set a language for viewing menus. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).<br><br>For more information, see the <b>Ektron CMS400.NET</b> User Manual section "Working with Menus" > "Managing Menus" > "Working with Menus in a Multi-Language System"                                                                                                                                                                                                           | Integer   |
| LaunchLinksinNew Window | If you set to <b>true</b> and a site visitor selects a menu option, it appears in a new browser window.<br><br>If <b>false</b> , the new page replaces the current page in the browser.                                                                                                                                                                                                                                                                                                                                       | Boolean   |
| ShowRootFolder          | If you set to <b>true</b> , a menu option listing the title of the root folder appears. If you set to <b>false</b> , the title of the root folder does not appear.                                                                                                                                                                                                                                                                                                                                                            | Boolean   |

| Property         | Value                                                                                                                                                                                                                                                                                                                                                                                     | Data Type |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| SingleEditButton | <p>Set to <b>true</b> if you want a single edit button (and no add button) to appear on the menu when you are logged in.</p> <p>If you set to <b>false</b>, <b>Edit Menu</b> and <b>Add Menu</b> options appear at the bottom of every menu and submenu.</p> <p>Example of SingleEditButton = true</p>  | Boolean   |

| Property                        | Value                                                                                                                                                                                                                 | Data Type |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| SingleEditButton<br>(continued) | <p>Example of SingleEditButton = false</p>                                                                                         | Boolean   |
| StartCollapsed                  | If you set to <b>true</b> , all submenus are closed when the menu initially displays.                                                                                                                                 | Boolean   |
| Start With RootCollapsed        | <p>If you set to <b>true</b>, the menu initially displays with its root menu closed.</p> <hr/> <p><u>Note: This property is ignored if the <code>ShowRootFolder</code> property is set to <code>false</code>.</u></p> | Boolean   |

| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Data Type |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Stylesheet          | <p>Enter the style sheet that will determine the appearance of the menus. Menu style sheets reside in the <i>site root/workarea/csslib/smartmenu</i> folder. You can modify any standard style sheet or create your own.</p> <p><b>Creating a Custom Style Sheet</b></p> <p>To support more than one SmartMenu server control per page, the CSS file's name is used as a class and is prefixed to all selectors (without the path or extension). The server control wraps the entire SmartMenu in a <code>&lt;div&gt;</code> tag, whose class name is derived from the filename.</p> <p>Only the base ASCII character set is supported. Also, spaces and periods before the extension are converted to underscores, since these are not permitted within the selector.</p> <p>So, to create your own style sheet, each selector must begin with the style sheet name. For example, <code>.my_custom_style_sheet .ekmenu</code>.</p> | String    |
| SuppressAddEdit     | <p>When set to True, suppress the Add and Edit buttons on the menu when a user is logged in to Ektron CMS400.NET. The default is False.</p> <p>True - suppress the Add and Edit button when a user is logged in to Ektron CMS400.NET.</p> <p>False - show the Add and Edit buttons when a user is logged in to Ektron CMS400.NET.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Boolean   |
| SuppressWrapperTags | <p>Suppresses the output of the span/div tags around the control. The default is False.</p> <p>True - Suppress wrap tags.</p> <p>False - Allow wrap tags.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Boolean   |
| WrapTag             | <p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - Use the <code>&lt;span&gt;</code> tag to designate an inline portion of an HTML document as a span element.</p> <p><b>Div</b> - Use the <code>&lt;div&gt;</code> tag used to apply attributes to a block of code.</p> <p><b>Custom</b> -Use a custom tag.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | String    |

## Section 508 Compliance Features of Smart Menus

**Ektron CMS400.NET's** Smart Menus are 508 compliant because they adhere to the following principles. They

- **are accessible** - the navigation functionality is accessible to all users who interact with a text-based Web page.
- **can work without DHTML** - Underneath the surface, Smart Menus are rendered as an unordered list. By default, they are styled with CSS and made dynamic with JavaScript. But, if you don't use these features, the menu still works although its look and feel are different.
- **are 100% navigable via the keyboard; no mouse actions are required** - Each Smart Menu has an <H2> header, which is used by some screen readers (such as Jaws) to assist visually impaired users with page navigation. A list of all headers is available via a key sequence.  
Also, all menu items are wrapped in links, which allow screen readers to assist a visually impaired user with page navigation. A list of links is available via a key sequence.
- **do not rely on color to convey important information** - the standard style sheets use underlining and bold to (in addition to color) to distinguish normal from selected menu options.  
Also, “**(Selected)**” follows the title of selected items. This text is suppressed when viewed with graphical browsers (such as IE6 and FireFox1.5) but is visible in text browsers and announced when a screen reader is used.
- **provide sufficient contrast** - the standard style sheets emphasize contrasting colors. In addition, because of the flexibility of Smart Menus, you can use your creative and artistic resources to develop menus that are attractive and have proper contrast.
- **are understandable even when the style sheet is turned off**
- **let a screen reader skip repetitive navigation links**
- **when a user clicks a menu option, the screen reader proceeds to the new page** - it does not read the remaining menu items.

---

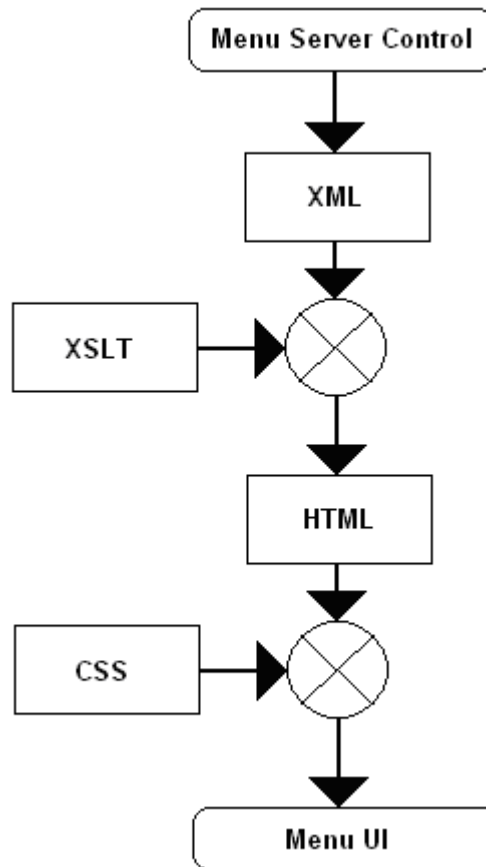
**NOTE**

To be sure your menus are 508 compliant, use either the [treemenuex.css](#) or [treemenuex\\_images.css](#) style sheet.

---

# Flexible Menu Server Control

The FlexMenu server control displays a menu on a Web form. Although it looks and acts like a Smart Menu, its architecture is substantially different. Specifically, a FlexMenu creates XML. So, you can modify its behavior using an XSLT file, and change its appearance using a cascading style sheet (.css) file. This design is illustrated below.



**NOTE** Before you can use this server control, you must create one or more menus in the **Ektron CMS400.NET** Workarea. To learn about creating menus, see "Working with Menus" in the **Ektron CMS400.NET** User Manual and "Menu Feature" in the **Ektron CMS400.NET** Administrator Manual.

This subsection contains the following topics.

- "Sample Menus" on page 233

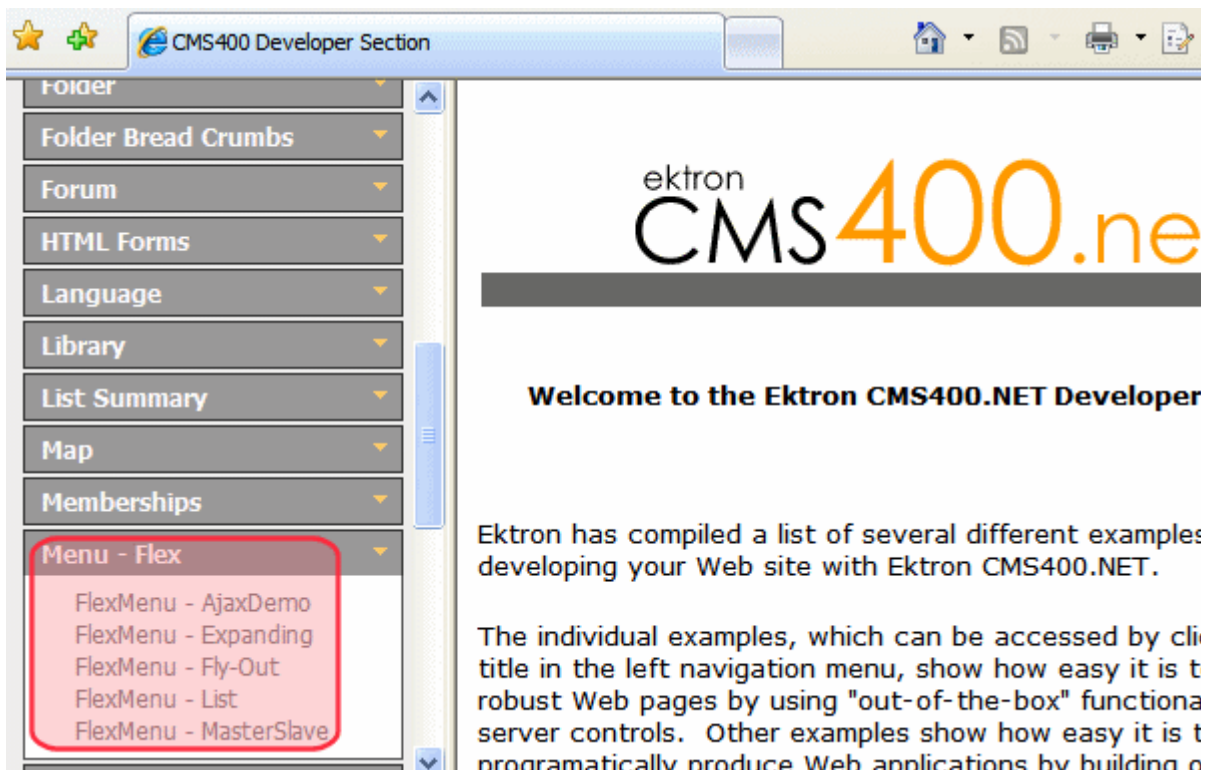
- "Working with the Flex Menu Xslt File" on page 238
- "Setting up a Master/Slave Menu Relationship" on page 239
- "Flex Menu Server Control Properties" on page 240

See Also: "Contrasting Menu Server Controls" on page 223

## Sample Menus

To facilitate the deployment of Flex menus, **Ektron CMS400.NET** provides several sample menus that show their capabilities. You can use whichever sample meets your needs, then modify it for your site.

The sample menus can be viewed from the **Developer Sample Page** (*siteroot/Developer/default.aspx*) > **Menu - Flex**.

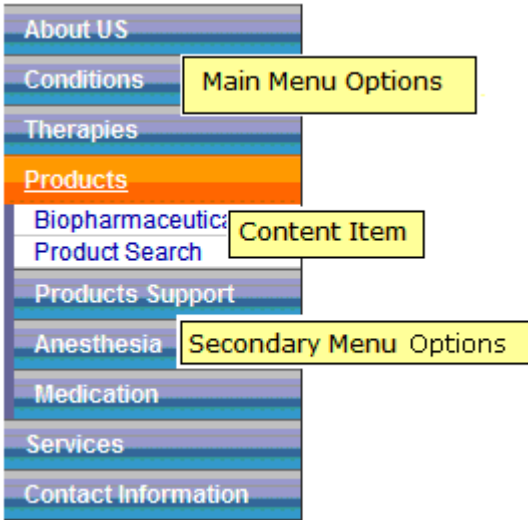


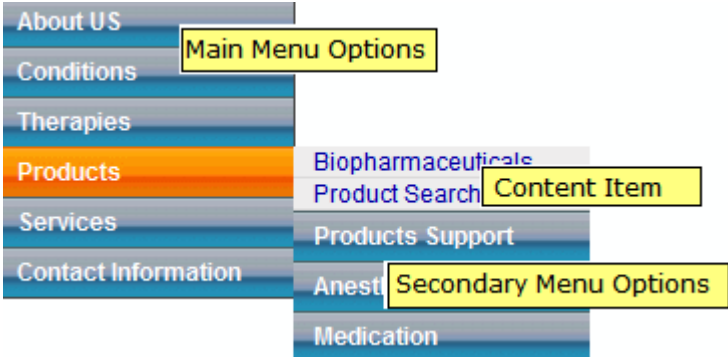
The samples are installed to the *siteroot/developer/menu/flexmenu* folder. Each sample has its own subfolder that contains sample aspx pages, a css file, and an xsl file.

## Description of the Sample Menus

| Sample Menu | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Location on Server                                                         |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| AjaxDemo    | <p>The only sample menu for which Ajax is enabled by default. Ajax provides a quicker response time because the client only loads menu levels on an “as needed” basis. This feature eliminates the need to do a page refresh to get new content. You may not notice the speed on small menus, but with large menus, the speed improvement is significant.</p> <hr/> <p><b>Note:</b> The first time an Ajax-enabled menu is invoked, it is slow because the server is building it. After that, its response time should be quicker than non-Ajax menus.</p> <hr/> <p><b>How the Menu Works</b></p> <ul style="list-style-type: none"> <li>• When you first click a menu, all text links are black. When you click it a second time, they are grayed out.</li> <li>• Secondary menus do not change.</li> <li>• There’s no way to close a menu after it opens.</li> </ul> <p><b>About US</b></p> <ul style="list-style-type: none"> <li>› Why Choose Ektron?</li> <li>› Business Practices</li> <li>› News <b>Content Item</b></li> <li>› Staff List</li> <li>› Ektron Medical Blog</li> <li>› Forum</li> <li>› Site Search</li> <li>› Using the Ektron Explorer</li> <li>› <b>Company Profile</b></li> <li>› <b>Careers</b> <b>Secondary Menu Options</b></li> </ul> <p><b>Conditions</b></p> <p><b>Therapies</b></p> <p><b>Products</b> <b>Main Menu Options</b></p> <p><b>Services</b></p> <p><b>Contact Information</b></p> | <p>siteroot/Developer/Menu/<br/>FlexMenu/AjaxDemo/<br/>FMAjaxDemo.aspx</p> |

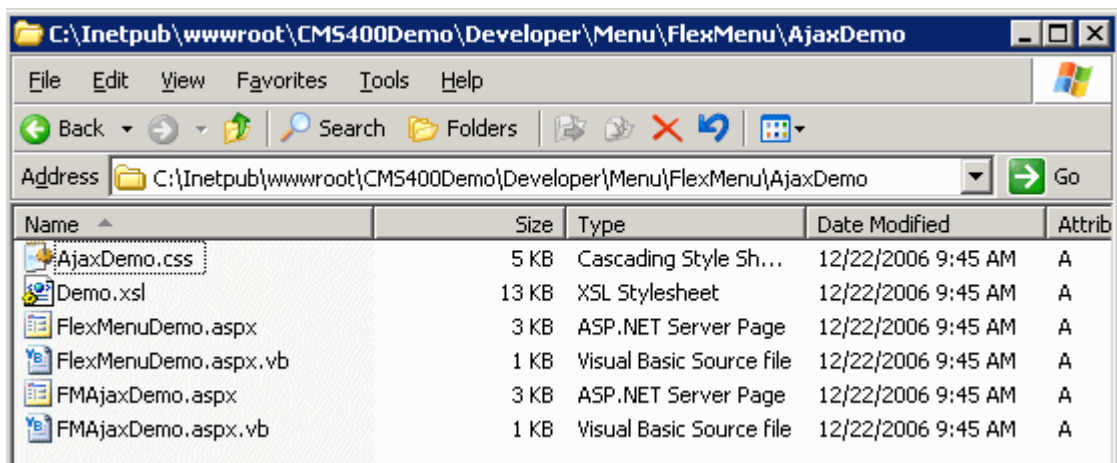


| Sample Menu | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Location on Server                                                                   |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| Expanding   | <ul style="list-style-type: none"> <li>Menu has gradient background, which changes when you hover over it.</li> <li>Menu options that display content have a plain background. Submenus look like main menus but are indented.</li> </ul>  <ul style="list-style-type: none"> <li>When you first click a menu, it expands vertically. When you click it a second time, it closes, so it looks like it did when you first viewed it.</li> </ul> | <code>siteroot//Developer/Menu/<br/>FlexMenu/Expanding/<br/>FMExpandDemo.aspx</code> |
| Fly-Out     | <ul style="list-style-type: none"> <li>Menus have gradient background. The color changes when you hover over it.</li> <li>Menu options that display content have a plain background. Their color changes when you hover over it. Submenus look like main menus but are offset to the right. (See illustration below.)</li> <li>When you first click a menu, it expands horizontally to the right. When you click it a second time, it closes, so it appears as it did when you first viewed it.</li> </ul>                      | <code>siteroot/Developer/Menu/<br/>FlexMenu/FlyOut/<br/>FMFlyOutDemo.aspx</code>     |

| Sample Menu                                                                       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Location on Server                                                |
|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|
|  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                   |
| List                                                                              | <ul style="list-style-type: none"> <li>• This menu's appearance is similar to the Ajax-Demo but it doesn't use Ajax.</li> <li>• The menu is an unordered list.</li> <li>• When you first click a menu, all text links are black. When you click it a second time, it closes, so it appears as it did when you first viewed it.</li> </ul> <p><b>About US</b></p> <ul style="list-style-type: none"> <li>› Why Choose Ektron?</li> <li>› Business Practices</li> <li>› News <b>Content Item</b></li> <li>› Staff List</li> <li>› Ektron Medical Blog</li> <li>› Forum</li> <li>› Site Search</li> <li>› Using the Ektron Explorer</li> <li>› <a href="#">Company Profile</a></li> <li>› <a href="#">Careers</a> <b>Secondary Menu Options</b></li> </ul> <p><a href="#">Conditions</a></p> <p><a href="#">Therapies</a></p> <p><a href="#">Products</a> <b>Main Menu Options</b></p> <p><a href="#">Services</a></p> <p><a href="#">Contact Information</a></p> | <p>siteroot/Developer/Menu/FlexMenu/List/FMListDemo.aspx</p>      |
| MasterSlave                                                                       | <p>Connects a slave menu to a master menu. In the graphic below, the upper menu (master) shows the top two levels while the lower (slave) menu shows levels three and on.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <p>siteroot/Developer/Menu/FlexMenu/MasterSlave/FMMSDemo.aspx</p> |

| Sample Menu                                | Description | Location on Server |
|--------------------------------------------|-------------|--------------------|
| <p><b>master:</b></p> <p><b>slave:</b></p> |             |                    |
| XML Modify                                 |             |                    |

## Copying and Editing Sample Menus



To create your menu based on a sample, follow these steps.

1. Examine the sample menus and decide which one best meets your needs for a particular page. See ["Description of the Sample Menus" on page 234](#).
2. If you want to *create a new template* (.aspx page) for the menu, place the new template in a new folder. Then, copy all files in the sample menu folder to the new folder and rename each file. For example, change Ajaxdemo.aspx to myflexmenu.aspx, AjaxDemo.css to myflexmenu.css, and demo.xsl to myflexmenu.xsl.

To insert a FlexMenu server control onto an *existing template*, copy the .css and .xsl file into the folder that contains the template.

---

**IMPORTANT!** The renamed .css file name must be all lowercase.

---

3. Open the .aspx page. Update the **stylesheet** and **DisplayXslt** properties to refer to the new .css and .xsl files.  
If the files do not reside in the template's folder, indicate their relative path using the syntax where slash (/) indicates the site root.
4. Open the .css file. Update the parent class name to match the new file name. For example, if you copied the AjaxDemo.css file, it looks like this.

```
.AjaxDemo .ekflexmenu
{
 /* DIV; wraps the entire menu */
 /* (the outer-most non-user container element) */
 color: rgb(0, 0, 255);
 /*width: 148px;*/
 font-size: 12px;
 font-family: Sans-Serif;
}

.AjaxDemo .ekflexmenu_submenu,
.AjaxDemo .ekflexmenu_submenu_hover,
.AjaxDemo .ekflexmenu_submenu_parent,
.AjaxDemo .ekflexmenu_submenu_parent_hover
{
 /* DIV; holds submenu items, such as a submenu title :
 top: 0px;
 left: 0px;
}

.AjaxDemo UL
{
 /* UL; a container for menu list items (useful for no
 /*list-style: none;
 /*margin: 0px;
 padding: 0px;*/
}
```

Notice that every class begins with the file name, **.AjaxDemo**. Replace all occurrences of this with the new file name. To continue the above example, begin all classes with **.myflexmenu**.

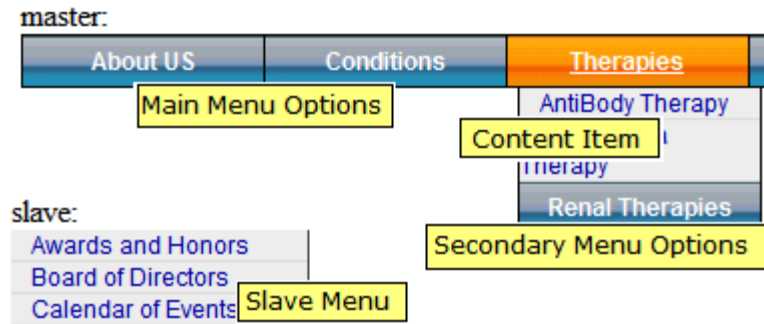
## Working with the Flex Menu Xslt File

This section explains some non-intuitive elements of the \*.xslt file.

| Element      | Context                                                                                                                   | Description                                                                                                                                                                                                                                                                                                                                                |
|--------------|---------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MenuFragment | <code>&lt;xsl:when test="/MenuDataResult/Info/menuFragment='false' "&gt;</code>                                           | A flag that indicates the XML data is not complete. Instead, it's a fragment that begins deeper than the top level (for example, a submenu fragment).<br><br>Because the data is incomplete, the XSLT processes the fragment differently. For example, don't generate Javascript startup code.                                                             |
| menuConst    | <code>&lt;xsl:attribute name="id"&gt;&lt;xsl:value-of select="\$menuConst" /&gt;0_ekflexmenu&lt;/xsl:attribute&gt;</code> | Each menu generates several elements, which the client code (Javascript) accesses via a unique ID. For example, Javascript needs to identify the selected submenu or item when a user clicks on an element.<br><br>menuConst is only used is when creating elements without a corresponding XML block, such as when creating a structure to hold the menu. |
| #NoScroll    | <code>&lt;xsl:attribute name="href"&gt;#NoScroll&lt;/xsl:attribute&gt;</code>                                             | NoScroll is sent to the href portion of a link when there is nothing to put there (for example, when the link is supposed to run Javascript).<br><br>NoScroll prevents the page from refreshing, going to another page, or scrolling when it should not.                                                                                                   |
| event        | <code>&lt;xsl:attribute name="onkeydown"&gt;return ekFlexMenu.menuBtnKeyHdlr(event);&lt;/xsl:attribute&gt;</code>         | To make the xslt cross-browser compatible, it must support different methods of obtaining/passing the event object. In this example, the global event object is passed to the handling function.<br><br><u>Note: event corresponds to window.event. window. is implied.</u>                                                                                |

## Setting up a Master/Slave Menu Relationship

A master/slave menu relationship is one in which a few top-level menu controls appear on the *master* menu while all remaining menu levels appears on a *slave* menu (see graphic below).



The best way to set up a master/slave menu relationship is to copy and edit the sample, `siteroot/Developer/Menu/FlexMenu/MasterSlave/FMMSDemo.aspx`.

Whether you modify that sample or create your own from scratch, follow these guidelines.

- Ektron recommends starting with the .xsl files provided with the Master/Slave sample (FM MasterDemo.xsl and FM SlaveDemo.xsl), since they are optimized to work with this kind of menu relationship.
- Place both menu server controls (master and slave) on the same Web form (.aspx page).
- Both server controls must refer to the same DefaultMenuID.
  - Give each server control a unique ID number. (This property is located under the **Misc** properties, not Ektron.)
- For the master menu
  - limit the `MenuDepth` property to the number of levels that will appear. In the above example, the master menu's depth is set to **2**.
  - set the `StartLevel` to 0 or 1, since you want the highest level menu options to appear
- For the slave menu
  - set the `MenuDepth` property to zero (0). The slave menu should show all lower menu levels.
  - set the `StartLevel` property to one number higher than the master menu's `MenuDepth` property. In this way, the slave menu takes over where the master menu leaves off.
  - set the `MasterControlID` property to the ID number of the master menu. (This property is located under the **Misc** properties, not Ektron.)

## Flex Menu Server Control Properties

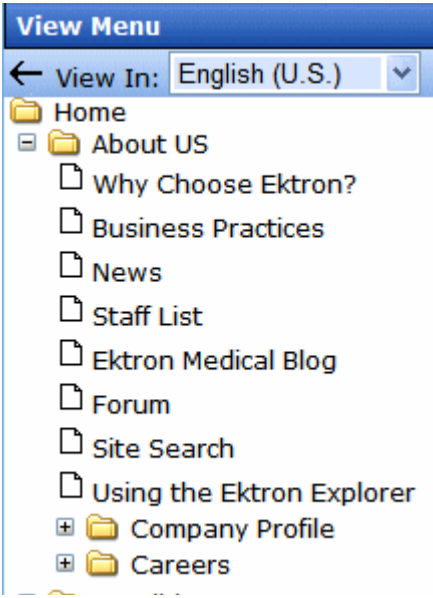
The Flex Menu server control properties are described in this table.

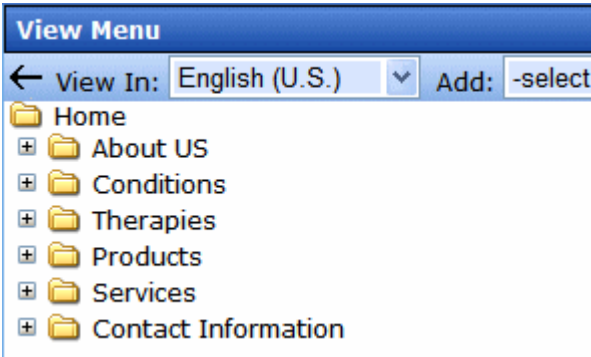
**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

| Property             | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Data Type |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Authenticated        | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc.<br><i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                                                                                                                                                                                               | Boolean   |
| AutoCollapseBranches | If you set to <b>true</b> , whenever a new submenu opens, all other submenus close.<br>If <b>false</b> , other submenus remain open when a new one opens.                                                                                                                                                                                                                                                                                                                         | Boolean   |
| CacheInterval        | Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). <i>See Also:</i> <a href="#">"Caching with Server Controls" on page 42</a>                                                                                                         | Integer   |
| DefaultMenuID        | The ID of a menu that appears where you insert this server control if no other menu is identified or available.<br>If you don't know the ID number of the menu, use the CMS Explorer to browse to it.<br><i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                                                                                                   | Integer   |
| DisplayXslt          | The xsl file used to render the menu. If nothing is specified, the menu is output as raw XML.<br>Flex menus are designed to use the xsl file to control the menu's behavior, and the .css file to control its display.<br><b>Ektron CMS400.NET</b> provides several sample menus, and each has an xslt file. If this is a new menu, you may find it easier to copy and edit an xslt file provided with a sample menu. <i>See Also:</i> <a href="#">"Sample Menus" on page 233</a> | String    |
| EnableAjax           | Set to <b>true</b> to enable Ajax, which only downloads submenus as needed. <i>See Also:</i> <a href="#">"AjaxDemo" on page 234</a>                                                                                                                                                                                                                                                                                                                                               | Boolean   |

| Property             | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Data Type |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| EnableMouseOverPopup | <p>If you set to <b>true</b>, submenus appear as soon as the cursor moves over them.</p> <p>If you set to <b>false</b>, submenus only appear if a site visitor clicks them or a keyboard equivalent.</p>                                                                                                                                                                                                                                                                                                              | Boolean   |
| EnableSmartOpen      | <p>If you set to <b>true</b>, any submenu on the Smart Menu can be set to open automatically. To learn about how to do this, see the <b>Ektron CMS400.NET</b> Administrator Manual section "Menu Feature" &gt; "Assigning Folders to a Menu."</p> <p>If you set to <b>false</b>, even if all required conditions are present, submenus on a Smart Menu do not automatically open.</p> <p>This property lets you prevent submenus from opening by default. Under some circumstances, such submenus look cluttered.</p> | Boolean   |
| MasterControlId      | <p>Only use this property if you are setting up a master/slave menu relationship, and this menu is the slave. If both are true, enter the ID of the master menu.</p> <p>See Also: <a href="#">""Setting up a Master/Slave Menu Relationship" on page 239</a></p>                                                                                                                                                                                                                                                      | String    |



| Property       | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Data Type |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| MenuDepth      | <p>To let site visitors browse through all menu levels, enter zero (0).</p> <p>To restrict site visitors to a menu level, enter the number of the lowest level.</p> <p>In the following example, if you set this property to <b>1</b>, a site visitor can browse through the <b>About Us</b> menu options but would not see the level 2 options (<b>Company Profile</b> and <b>Careers</b>).</p>  | Integer   |
| StartCollapsed | If you set to <b>true</b> , when the menu first appears, all submenus are closed.                                                                                                                                                                                                                                                                                                                                                                                                   | Boolean   |

| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Data Type |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| StartLevel          | <p>Enter a number to indicate the level at which you want this menu to display when it first appears. To begin the menu display at the root level, enter zero (0).</p> <p>In the following example, the <b>Home</b> folder is level 0. The others are level 1.</p>  <p>A site visitor can click a menu option to navigate to folders below the displayed level.</p> | Integer   |
| Stylesheet          | <p>Enter the style sheet that will determine the appearance of the menus.</p> <p>Flex menus use an .xsl file to control their behavior, and a .css file to control their display.</p> <p><b>Ektron CMS400.NET</b> provides several sample menus, and each has a .css file. If this is a new menu, you may find it easier to copy and edit a .css file provided with a sample menu. See Also: <a href="#">"Sample Menus" on page 233</a></p>           | String    |
| SuppressAddEdit     | <p>When set to True, suppress the Add and Edit buttons on the menu when a user is logged in to Ektron CMS400.NET. The default is False.</p> <p><b>True</b> - suppress the Add and Edit button when a user is logged in to Ektron CMS400.NET.</p> <p><b>False</b> - show the Add and Edit buttons when a user is logged in to Ektron CMS400.NET.</p>                                                                                                   | Boolean   |
| SuppressWrapperTags | <p>This property is set to <code>false</code> because Ajax uses <code>&lt;div&gt;</code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code>.</p>                                                                                                                                                                                                                                                      | Boolean   |

| Property | Value                                                                                                                                                                                                                                                                                                                                                                    | Data Type |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| WrapTag  | <p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - Use the <code>&lt;span&gt;</code> tag to designate an inline portion of an HTML document as a span element.</p> <p><b>Div</b> - Use the <code>&lt;div&gt;</code> tag used to apply attributes to a block of code.</p> <p><b>Custom</b> - Use a custom tag.</p> | String    |

# Menu Server Control

The Menu server control calls a menu to be displayed on a page. Using the Menu server control, you can manipulate a menu by using the DisplayXslt property. Below is a menu display with the SampleMenu Xslt.



Before you can use the Menu server control, you must create a menu in the **Ektron CMS400.NET** work area. To learn about creating menus, see “Working with Menus” in the **Ektron CMS400.NET** User Manual.

This subsection contains the following topics:

- ["Menu Server Control Properties" on page 246](#)
- ["Using DisplayXslt Samples" on page 248](#)
- ["Retrieving the XML Structure of a Menu" on page 252](#)

See Also: ["Contrasting Menu Server Controls" on page 223](#)

## Menu Server Control Properties

The Menu server control properties are described in this table.

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

| Property      | Value                                                                                                                                                                                                               | Data Type |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Authenticated | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc.<br><i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a> | String    |

| Property         | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Data Type |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| CacheInterval    | Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>                                                                                                                                                   | Integer   |
| DefaultMenuID    | The ID of a menu that appears where you insert this server control if no other menu is identified or available. If you don't know the ID number of the menu, use the CMS Explorer to browse to it.<br><br>See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                                                                                                                                             | Integer   |
| DisplayXslt      | The Xslt to use to render the menu.<br><b>None</b> - databind only<br><b>SampleMenu</b> - A sample display, formatted as a bulleted menu list<br><b>TreeMenu</b> - A sample display, formatted as a folder tree. You can expand the tree by clicking on the folder icon.<br><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page<br><br>For more information on using the SampleMenu and TreeMenu DisplayXslt, see <a href="#">"Using DisplayXslt Samples" on page 248.</a> | String    |
| DynamicParameter | Select <b>menu_id</b> . When you do, this server control uses the menu passed as a URL parameter.                                                                                                                                                                                                                                                                                                                                                                                                                     | String    |
| Hide             | Used to hide a menu in design time and run time.<br><b>True</b> = Hide menu<br><b>False</b> = Show menu                                                                                                                                                                                                                                                                                                                                                                                                               | Boolean   |
| Language         | Set a language for viewing menus. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).<br><br>For more information, see the <b>Ektron CMS400.NET</b> User Manual section "Working with Menus" > "Managing Menus" > "Working with Menus in a Multi-Language System"                                                                                                                                                                                                   | Integer   |

| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                 | Data Type |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Stylesheet          | Specify the location and style sheet for the menu. For example: <code>\Workarea\csslib\mytest.css</code> .<br>Leave blank to use the default.<br><code>global.css</code> is the default style sheet. It is located in <code>\webroot\siteroot\Workarea\csslib</code> . For more information on style sheets, see <a href="#">"Using a Style Sheet" on page 260</a> .                  | String    |
| SuppressAddEdit     | When set to True, suppress the Add and Edit buttons on the menu when a user is logged in to Ektron CMS400.NET. The default is False.<br><b>True</b> - suppress the Add and Edit button when a user is logged in to Ektron CMS400.NET.<br><b>False</b> - show the Add and Edit buttons when a user is logged in to Ektron CMS400.NET.                                                  | Boolean   |
| SuppressWrapperTags | Suppresses the output of the span/div tags around the control. The default is <b>False</b> .<br><b>True</b> - Suppress wrap tags.<br><b>False</b> - Allow wrap tags.                                                                                                                                                                                                                  | Boolean   |
| WrapTag             | Allows a developer to specify a server control's tag. The default is <b>Span</b> .<br><b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an inline portion of an HTML document as a span element.<br><b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.<br><b>Custom</b> - Allows you to use a custom tag. | String    |

## Using DisplayXslt Samples

Two DisplayXslt samples are provided with the Menu server control, SampleMenu and TreeMenu. This section explains how to use them.

### SampleMenu

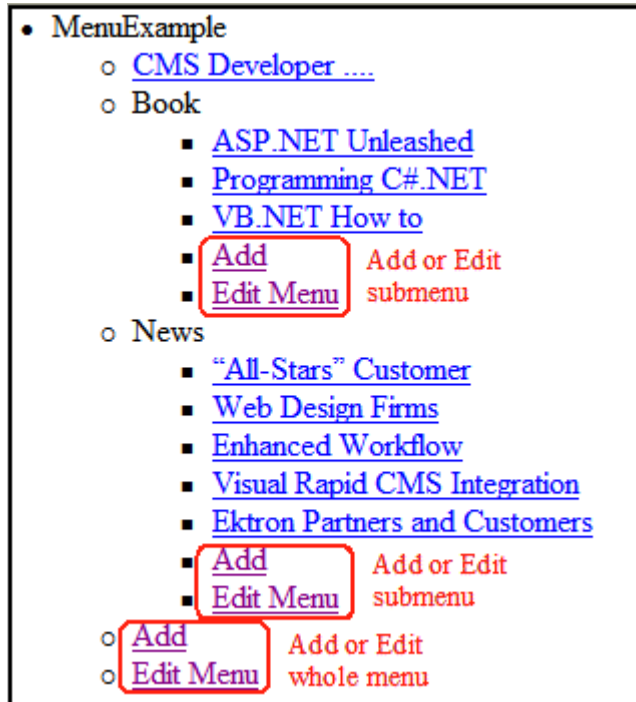
The SampleMenu DisplayXslt lets you display an **Ektron CMS400.NET** menu as a bulleted item list. See below for an example.

- MenuExample
  - [CMS Developer ....](#)
  - Book
    - [ASP.NET Unleashed](#)
    - [Programming C#.NET](#)
    - [VB.NET How to](#)
  - News
    - [“All-Stars” Customer](#)
    - [Web Design Firms](#)
    - [Enhanced Workflow](#)
    - [Visual Rapid CMS Integration](#)
    - [Ektron Partners and Customers](#)

When you are logged in to your **Ektron CMS400.NET** site, two menu items are added to each section of the menu: Add and Edit Menu. The user can use these to add a new menu item or edit an existing one. See example below.

For Information on adding a menu item, see **Ektron CMS400.NET** Users Manual > “Working with Menus” > “Managing Menus” > “Adding a New Content Block to a Menu via Navigation Link on a Web Page”.

For Information on editing a menu, see **Ektron CMS400.NET** Users Manual > “Working with Menus” > “Managing Menus” > “Editing a Menu via Navigation Link on a Web Page”.



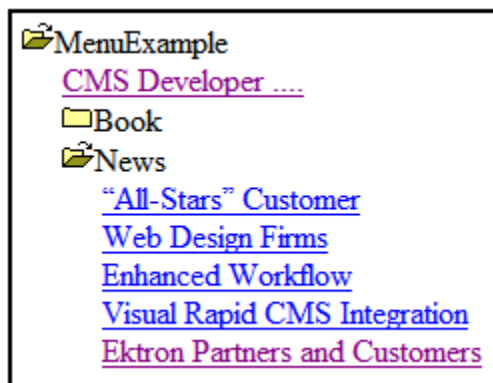
### SampleMenu XSL code

For an example of the XSL code used in the SampleMenu, click the following link and choose the appropriate version.

<http://www.ektron.com/developers/cms400kb.cfm?id=2114>

### TreeMenu

The TreeMenu DisplayXslt lets you display an **Ektron CMS400.NET** menu as a clickable folder list. You can expand and collapse the menu by clicking on the folder icons. See below for an example.

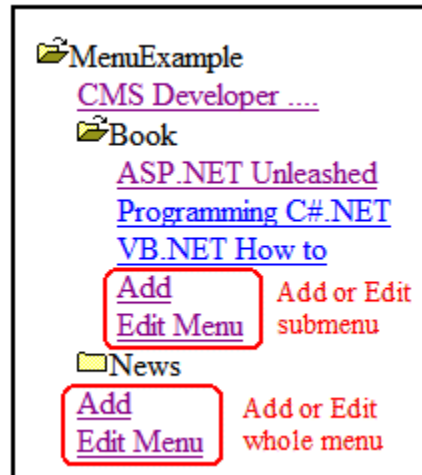




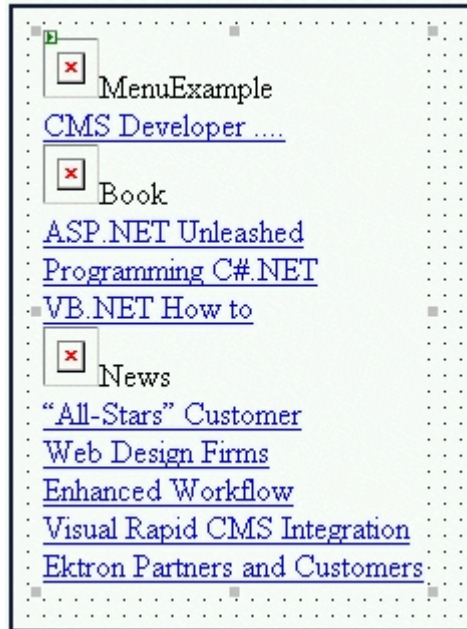
When you are logged in to your **Ektron CMS400.NET** site, two menu items are added to each section of the menu: Add and Edit Menu. The user can use these to add a new menu item, or edit an existing menu item. See example below.

For Information on a adding a menu Item, see **Ektron CMS400.NET** Users Manual > “Working with Menus” > “Managing Menus” > “Adding a New Content Block to a Menu via Navigation Link on a Web Page”.

For Information on a editing a menu, see **Ektron CMS400.NET** Users Manual > “Working with Menus” > “Managing Menus” > “Editing a Menu via Navigation Link on a Web Page”.



**NOTE** When using the TreeMenu server control with Visual Studio and Windows 2003, the folder images do not display during design-time. They do, however, display correctly during run-time.



### TreeMenu XSL code

For an example of the XSL code used in the MenuTree, click the following link and choose the appropriate version.

<http://www.ektron.com/developers/cms400kb.cfm?id=2114>

## Retrieving the XML Structure of a Menu

Retrieving the XML structure of XML content allows for greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a Menu server control onto it.
3. Set the DefaultMenuID properties.
4. Drag and drop a Textbox on the Web form.
5. Set the TextMode property to MultiLine.

**NOTE** It is also recommended that you set the width of the text box to at least 400px.

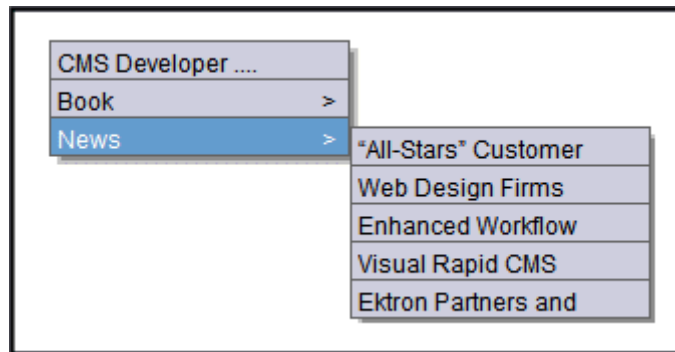
6. On the code behind page, add the following line.  
`Textbox1.Text = Menu1.XmlDoc.InnerXml`
7. Build the project.

8. View the Web form in a browser.
9. The XML structure of the menu appears in the textbox.

# DhtmlMenu Server Control

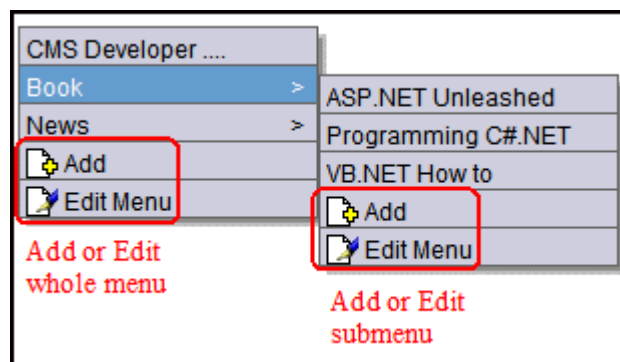
**NOTE** For optimal performance, Ektron recommends using at minimum Internet Explorer 5.5, Netscape 6.0, or Mozilla Firefox 1.0.

The DhtmlMenu server control lets you display a menu with drop down boxes and submenus. This lets developers add a menu to a page without taking up a lot space. You can modify the menu's style properties via a style sheet.



Before using the DhtmlMenu server control, you must create menus in the Ektron CMS400.NET workarea. See "Working with Menus" in the CMS400.NET User Manual.

The DhtmlMenu server control also lets logged in users add content to a menu. When you are logged in to your **Ektron CMS400.NET** site, two items are added to each menu: **Add** and **Edit Menu**. Use them to add a new menu item or edit an existing one. See example below.



For Information on a adding a menu item, see **Ektron CMS400.NET** Users Manual > "Working with Menus" > "Managing Menus" > "Adding a New Content Block to a Menu via Navigation Link on a Web Page".

For Information on a editing a menu, see **Ektron CMS400.NET** Users Manual > “Working with Menus” > “Managing Menus” > “Editing a Menu via Navigation Link on a Web Page”.

**WARNING!**

You cannot create a DHTML menu using code behind -- you must drag and drop it. Once a DHTML menu is on a Web form, you can manipulate its properties using the code behind.

This subsection contains the following topics:

- ["DhtmlMenu Server Control Properties" on page 255](#)
- ["Using a Style Sheet" on page 260](#)
- ["Formatting the DhtmlMenu Server Control" on page 260](#)

See Also: ["Contrasting Menu Server Controls" on page 223](#)

## DhtmlMenu Server Control Properties

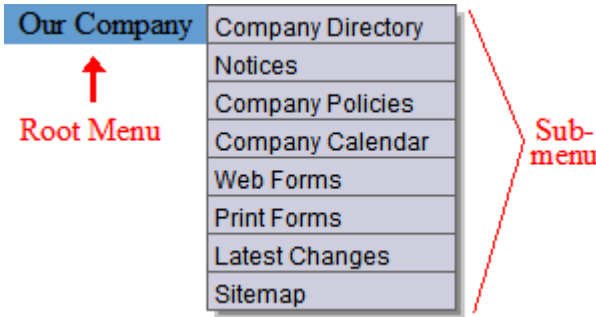
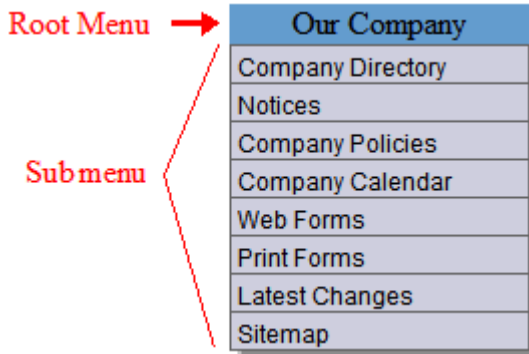
The DhtmlMenu server control properties are described in this table.


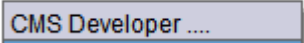

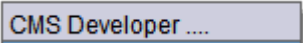
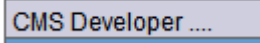
**NOTE**

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

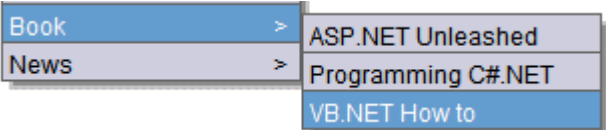
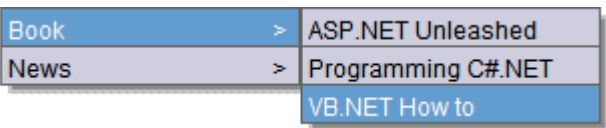
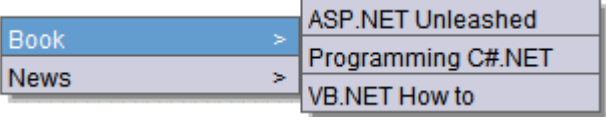
| Property      | Value                                                                                                                                                                                                                                                                                                                                  | Data Type |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Align         | Specifies the horizontal alignment of the root menu text.<br>The default is <b>Center</b> .<br><b>NotSet</b> - Use the default setting.<br><b>Left</b> - Aligns text to the left.<br><b>Center</b> - Aligns text to the center.<br><b>Right</b> - Aligns text to the right.<br><b>Justify</b> - Aligns text justified to both margins. | String    |
| Authenticated | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <a href="#">Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>                                                                                                                              | String    |

| Property          | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Data Type |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| CssClass          | <p>Apply a .css class to the root menu. Enter the class name from the .css defined in the Stylesheet property. See Also: <a href="#">"Stylesheet" on page 258</a></p> <p>Since pop_style.css is the default style sheet, you can use its classes without defining it in the Stylesheet property.</p> <p>For example, if you enter lowtext in the property, the root menu changes as shown below.</p> <div> <div>No CssClass applied</div> <div>lowText CssClass applied</div> </div> <div> <div>MenuExample</div> <div>MenuExample</div> </div> | String    |
| DefaultMenuID     | <p>The ID of a menu that appears where you inserted this server control if no other menu is identified or is available.</p> <p>If you don't know the ID number of the menu, use the CMS Explorer to browse to it. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a></p>                                                                                                                                                                                                                           | Integer   |
| DynamicParameter  | Select <b>menu_id</b> . When you do, this server control uses the menu passed as a URL parameter.                                                                                                                                                                                                                                                                                                                                                                                                                                               | String    |
| EnableIE6FormsFix | This should only be enabled when you have a DHTML menu rendering over an HTML select element. In IE6, the DHTML renders below the drop down. This temporarily disables the element when the DHTML Menu is rendered.                                                                                                                                                                                                                                                                                                                             | Boolean   |
| Hide              | <p>Used to hide a menu in design time and run time.</p> <p><b>True</b> = Hide menu</p> <p><b>False</b> = Show menu</p>                                                                                                                                                                                                                                                                                                                                                                                                                          | Boolean   |

| Property           | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Data Type |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| HorizontalDisplay  | <p>Displays submenu to the right of the root menu. This only affects the first submenu of the root menu. All subsequent submenus appear to the right of each submenu.</p> <p>The default is <b>False</b></p> <p><b>True</b> = Submenu appears to the right of the root menu</p>  <p><b>False</b> = Submenu appears below the root menu</p>  | Boolean   |
| Language           | <p>Set a language for viewing Dhtml Menus. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p> <p>For more information, see the <b>Ektron CMS400.NET</b> User Manual section “Working with Menus” &gt; “Managing Menus” &gt; “Working with Menus in a Multi-Language System”</p>                                                                                                                                                                           | Integer   |
| MenuHoverOverColor | Sets the background color for the root menu when hovering over it.                                                                                                                                                                                                                                                                                                                                                                                                                                               | String    |

| Property              | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Data Type |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| MenuStyle             | <p>A specified inline style that defines the root menu. For example:</p> <pre>font: bold 18px Arial;color:royalblue; background-color:slateblue</pre>                                                                                                                                                                                                                                                          | String    |
| Stylesheet            | <p>Specify the location and style sheet for the submenu. For example: \Workarea\csslib\mytest.css. Leave blank to use the default.</p> <p>pop_style.css is the default style sheet. It is located in \siteroot\Workarea\csslib. For more information on style sheets, see <a href="#">"Using a Style Sheet" on page 260</a>.</p> <p>You can use the Stylesheet property in conjunction with the CssClass property to define the root menu. See Also: <a href="#">"CssClass" on page 256</a></p> | String    |
| SubMenuBackColor      | Sets the background color for the submenu.                                                                                                                                                                                                                                                                                                                                                                                                                                                      | String    |
| SubMenuHoverOverColor | Sets the background color for the submenu when the cursor hovers over it.                                                                                                                                                                                                                                                                                                                                                                                                                       | String    |
| SubMenuItemHeight     | <p>Sets the height of each submenu item in pixels. The default is 20.</p> <p>Example set to 20: </p> <p>Example set to 30: </p>                                                                                                                                                                                         | Integer   |
| SubMenuItemWidth      | <p>Sets the width of each submenu item in pixels. The Default is 150</p> <p>Example set to 150: </p> <p>Example set to 130: </p>                                                                                                                                                                                         | Integer   |



| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Data Type |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| SubMenuTopOffset    | <p>Sets the location of the top of the submenu relative to its parent. This number, in pixels, refers to the number of pixels the submenu appears below the parent menu. The default is 5.</p> <p>Example set to 5:</p>  <p>Example set to 0 (zero):</p>  <p>Example set to -10:</p>  | Integer   |
| SuppressAddEdit     | <p>When set to True, suppress the Add and Edit buttons on the menu when a user is logged in to Ektron CMS400.NET. The default is False.</p> <p><b>True</b> - suppress the Add and Edit button when a user is logged in to Ektron CMS400.NET.</p> <p><b>False</b> - show the Add and Edit buttons when a user is logged in to Ektron CMS400.NET.</p>                                                                                                                                                                                         | Boolean   |
| SuppressWrapperTags | <p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags.</p> <p><b>False</b> - Allow wrap tags.</p>                                                                                                                                                                                                                                                                                                                                                          | Boolean   |

| Property | Value                                                                                                                                                                                                                                                                                                                                                                         | Data Type |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| WrapTag  | <p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an inline portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p> | String    |

## Using a Style Sheet

You can create your own style sheet, or modify the existing default one, `pop_style.css`. It is located in `\site root\Workarea\csslib`. For a description of the classes contained in `pop_style.css`, see "[pop\\_style.css](#)" on [page 940](#).

Style Sheets let you choose different text colors, text alignment, text fonts, and text size to create a customized menu.

### Best Practice

Ektron recommends copying and renaming the default style sheet. This insures you always have a clean file to start with. This also gives you something to reference if you are not getting the results you expect.

The style sheet must be located below the site root.

You can only define one menu style sheet for each Web page. If you have more than one menu on a single Web page, they share a style sheet.

For more information on style sheets, visit <http://www.w3.org/Style/CSS/>.

## Formatting the DhtmlMenu Server Control

Additional information on formatting the DhtmlMenu server control can be found in the following Knowledge Base article.

[http://dev.ektron.com/kb\\_article.aspx?id=2441](http://dev.ektron.com/kb_article.aspx?id=2441)

# MetaData Server Control

The Metadata server control lets you add the metadata from content blocks to a Web page. This lets developers add metadata quickly without having to type it in. You can add metadata from a single content block, multiple content blocks, or by dynamically passing a content ID from a URL.

This subsection contains the following topics:

- ["MetaData Server Control vs. MetaDataList Server Control" on page 261](#)
- ["MetaData Server Control Properties" on page 261](#)
- ["Using the Simple Dublin Core Metadata Standard" on page 263](#)
- ["Using the MetaData Server Control" on page 264](#)

## MetaData Server Control vs. MetaDataList Server Control

With the MetaData server control, you add metadata from content blocks to your Web page. With the MetaDataList server control, you create a list of content blocks to display on your site, based on the Metadata in each content block. For Information on the MetaDataList server control, see ["MetaDataList Server Control" on page 266](#).

## MetaData Server Control Properties

The Metadata server control properties are described in this table.

### NOTE

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

| Property      | Value                                                                                                                                                                                                     | Data Type |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| Authenticated | Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <a href="#">Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a> | String    |

| Property            | Value                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Data Type |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| DefaultContentID    | <p>The content block ID from which the server control gets the metadata.</p> <p>If you want to add metadata from several content blocks, set this property to 0 (zero) and use the DefaultItemList property to identify the content blocks.</p> <p>If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a></p>                                                 | Integer   |
| DefaultItemList     | <p>A comma delimited list of content block IDs from which to get metadata. DefaultContentID must be set to 0 (zero) so you can add IDs to the list.</p> <p><code>DefaultItemList</code> [12,4,7]</p>                                                                                                                                                                                                                                                                                               | Integer   |
| DynamicParameter    | Gets or sets the QueryString parameter to read a content ID dynamically.                                                                                                                                                                                                                                                                                                                                                                                                                           | String    |
| GenerateDublinCore  | <p>When enabled, this property automatically creates seven of the Simple Dublin Core metadata fields from standard CMS400.NET system properties. The default is false.</p> <p><b>True</b> = Generate Simple Dublin Core metadata fields<br/> <b>False</b> = Do not generate Simple Dublin Core metadata fields</p> <p>The seven fields and how they are associated with the CMS400.NET properties is explained in <a href="#">"Using the Simple Dublin Core Metadata Standard" on page 263</a></p> | Boolean   |
| Hide                | <p>Used to hide a Metadata server control in design time and run time.</p> <p><b>True</b> = Hide Metadata server control<br/> <b>False</b> = Show Metadata server control</p>                                                                                                                                                                                                                                                                                                                      | boolean   |
| Language            | Set a language for viewing form content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).                                                                                                                                                                                                                                                                                                                                                     | Integer   |
| SuppressWrapperTags | <p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags.<br/> <b>False</b> - Allow wrap tags.</p>                                                                                                                                                                                                                                                                                                                   | Boolean   |

| Property | Value                                                                                                                                                                                                                                                                                                                                                                          | Data Type |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| WrapTag  | <p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p> | String    |

## Using the Simple Dublin Core Metadata Standard

Simple Dublin Core is a set of fifteen standard names for metadata fields designed to cover the most useful items of information on a document. From the Dublin Core FAQ: Dublin Core metadata provides card catalog-like definitions for defining the properties of objects for Web-based resource discovery systems.

For more information on the Dublin Core standard refer to the Usage Guide: <http://www.dublincore.org/documents/usageguide/>

To generate Dublin Core metadata, set the GenerateDublinCore property to True. This creates seven of the fifteen Dublin Core metadata fields. These fields are automatically filled with the information from the equivalent Ektron CMS400.NET property. Below is a list of the seven fields that are created and their Ektron CMS400.NET equivalent property.

### IMPORTANT!

These seven Dublin Core fields are automatically populated with information from their equivalent Ektron CMS400.NET property.

| Dublin Core Field Name | Ektron CMS400.NET Property                               |
|------------------------|----------------------------------------------------------|
| DC.title               | Content block title                                      |
| DC.description         | Plain text version of a content block teaser (summary)   |
| DC.contributor         | Content block last editor name                           |
| DC.date                | Content block last edit date                             |
| DC.format              | "text/html"                                              |
| DC.identifier          | URL of current page (from ASP.NET Server.Request object) |

| Dublin Core Field Name | Ektron CMS400.NET Property                                                                         |
|------------------------|----------------------------------------------------------------------------------------------------|
| DC.language            | CMS language cookie / current site language, expressed as a .NET System.Globalization Culture Name |

To fully comply with the Simple Dublin Core metadata element set, the administrator must create the remaining eight Dublin Core fields as standard CMS400.NET Metadata fields and apply them to all CMS400.NET folders. Next, the CMS users must fill in the appropriate values for each content block.

For information on the remaining eight Simple Dublin Core fields, see the Administrator Manual section “Managing Content Folders” > “Metadata” > “Simple Dublin Core Metadata”.

## Using the MetaData Server Control

These steps show how to use the MetaData server control.

1. Drag a MetaData server control into a template.
2. Set the properties of the Metadata server control. This will create the following HTML in the HTML body.

```
<cms:MetaData id="MetaData1" runat="server" DefaultContentID="12"></cms:MetaData>
```

Or if you are using multiple content block IDs In the DefaultItemList, the following HTML will be created.

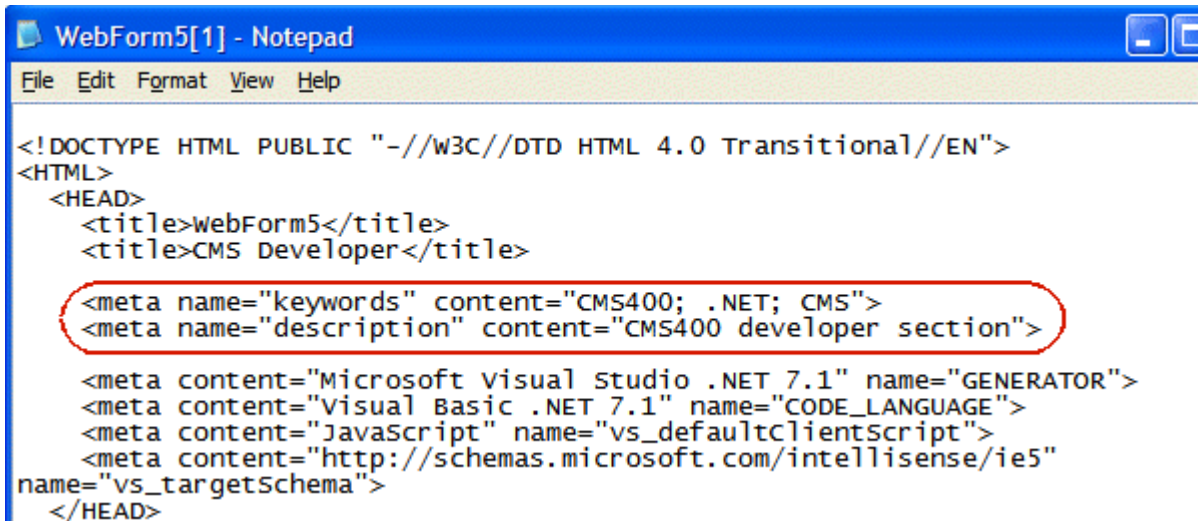
```
<cms:metadata id="MetaData1" runat="server" DefaultItemList="[12,7,4]"></cms:metadata>
```

3. Click on the HTML tab and copy that line from the <body> tag into the <head> tag.

```
<HEAD>
 <title>WebForm5</title>
 <cms:MetaData id="Metadata2" runat="server" DefaultContentID="12"></cms:MetaData>
 <meta name="GENERATOR" content="Microsoft Visual Studio .NET 7.1">
 <meta name="CODE_LANGUAGE" content="Visual Basic .NET 7.1">
 <meta name="vs_defaultClientScript" content="JavaScript">
 <meta name="vs_targetSchema" content="http://schemas.microsoft.com/intellisense/i
</HEAD>
<body>
 <form id="Form1" method="post" runat="server">
 </form>
</body>
```

4. Save the Web form and rebuild the solution.
5. Open the Web page in the browser.
6. Right click on the Web page and click **View Source**. Look in the head tag. The meta tags from the content block are added to the page.

The metadata information added is shown below.



```

WebForm5[1] - Notepad
File Edit Format View Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
 <HEAD>
 <title>webForm5</title>
 <title>CMS Developer</title>
 <meta name="keywords" content="CMS400; .NET; CMS">
 <meta name="description" content="CMS400 developer section">
 <meta content="Microsoft Visual Studio .NET 7.1" name="GENERATOR">
 <meta content="Visual Basic .NET 7.1" name="CODE_LANGUAGE">
 <meta content="JavaScript" name="vs_defaultClientScript">
 <meta content="http://schemas.microsoft.com/intellisense/ie5"
name="vs_targetSchema">
 </HEAD>

```

# MetaDataList Server Control

Use the MetaDataList server control to create lists based on Keyword Names and Keyword Values contained within the metadata of content. In the example below, KeyWordNames are circled in red, and KeyWordValues are circled in green.

Title: Contact Ektron [English (U.S.)]

**Content** **Summary** **Metadata** **Schedule** **Comment**

\* = Required fields

**Title:** (500 max.) Separator Character ";"

Contact Ektron

486 characters left

Default

**Keywords:** (500 max.) Separator Character ";"

email; support; questions; ektron; contact;

The list can display the information as a list of hyperlinks. You can choose, based on properties you set, to display the summary and how to order the display. For information on adding Metadata to a content block, see the **Ektron CMS400.NET User Manual** > "Adding or Editing Metadata".

This subsection contains the following topics:

- ["MetaData Server Control vs. MetaDataList Server Control" on page 266](#)
- ["MetaDataList Server Control Properties" on page 266](#)
- ["Retrieving the XML Structure of a MetaDataList" on page 275](#)

## MetaData Server Control vs. MetaDataList Server Control

With the MetaData server control, you add metadata from content blocks to your Web page. (See Also: ["MetaData Server Control" on page 261.](#)) With the MetaDataList server control, you create a list of content to display on your site based on the content's metadata.

## MetaDataList Server Control Properties

The MetaDataList server control properties are described in this table.



**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>	String
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a></p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>	Integer
ContentType	<p>One of the following</p> <ul style="list-style-type: none"> <li>• all types</li> <li>• archived content</li> <li>• archived forms</li> <li>• content</li> <li>• forms</li> </ul> <p>To learn about archived content, see the <b>Ektron CMS400.NET</b> User Manual, "Scheduling Content to Begin and End" &gt; "Setting Archive Options."</p>	String

Property	Value	Data Type
DisplayXslt	<p>Determines how the information on the page is displayed</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder</p> <p>See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary.</p> <p>See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String

Property	Value	Data Type
EnablePaging	<p>This property, in conjunction with the <code>MaxNumber</code> property, lets site visitors view an unlimited number of content items while controlling the amount of screen space. To accomplish this, the content display is limited to the number set in the <code>MaxNumber</code> property.</p> <p>If you set this property to <b>true</b>, and the number of content items exceeds the <code>MaxNumber</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="565 598 976 826" data-label="Image"> </div> <p>So, for example, if specified metadata is found in 9 items and the <code>MaxResults</code> property is set to 3, the screen displays only the first three items. When the site visitor clicks <b>[Next]</b>, he sees items 4, 5 and 6, etc.</p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to <b>True</b>, the <code>CacheInterval</code> property is disabled.</p> <hr/>	
ExactPhrase	<p>Determines whether the <code>KeywordValue</code> needs to match the metadata value exactly.</p> <p>For example, if "site" is the <code>KeywordValue</code>, the title of a content block is "Welcome to the site" and <code>ExactPhrase</code> is set to <b>true</b>, you would not see the content block in the metadata list. This is because "site" does not equal "Welcome to the site".</p> <p><b>True</b> = Match the exact phrase  <b>False</b> = Doesn't need to match exact phrase</p>	Boolean
FolderID	<p>The folder ID from which content is retrieved. At the <code>Recursive</code> property, you determine if content in this folder's subfolders is also retrieved.</p>	Integer

Property	Value	Data Type
GetHtml	Set to <b>True</b> if you want to display the content (html body) for all content to appear on this metadata list. For example, you want to display content inside a Web server control such as a GridView.	Boolean
Hide	Used to hide a metadata list in design time and run time. <b>True</b> = Hide metadata list <b>False</b> = Show metadata list	Boolean
Includelcons	<p>Choose whether to display icons next to the metadata list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$ImageIcon] variable is used in an EkML file.</p> <hr/> <div data-bbox="506 829 1035 1221" data-label="Image"> </div>	Boolean

Property	Value	Data Type
KeyWordName	<p>KeyWordName represents a metadata definition, that is, the container for the KeyWordValues. Examples of a KeyWordName are <b>Keywords</b> and <b>Title</b>.</p> <p>To view an illustration of the relationship between KeyWordName and KeyWordValues, see page 266.</p> <p>If you are authenticated, you can click the ellipsis button and select from a list of existing metadata definitions. See <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>.</p> <p>For information on creating metadata definitions, see the <b>Ektron CMS400.NET</b> Administrators Manual &gt; "Managing Content" &gt; "Metadata" &gt; "Managing Metadata Definitions" &gt; "Adding a Metadata Definitions"</p>	String
KeyWordValue	<p>Enter the values associated with the KeyWordName. Only content whose metadata (defined at the KeyWordName property) matches this value appears on the metadata list.</p> <p>Examples of a KeyWordValue are "home; page; company." To view an illustration of the relationship between KeyWordName and KeyWordValues, see page 266.</p> <hr/> <p><b>Note:</b> The character that separates multiple items is defined at the <code>KeyWordValueSeparator</code> property.</p> <hr/> <p><b>Note:</b> At the <code>KeyWordValueMatchAll</code> property, you determine if all metadata definition values must match or any one of them.</p> <hr/>	String

Property	Value	Data Type
KeyWordValueMatchAll	<p>This property is only used if you enter more than one keyword value.</p> <p>If you do, and only want content to appear on the metadata list if <i>all</i> values entered at the KeyWordValue field match its metadata values, enter <b>true</b>.</p> <p>If metadata can appear on the list as long as <i>any</i> value defined at the KeyWordValue field matches the selected metadata value for a content item, enter <b>false</b>.</p> <p><b>Example</b></p> <p>Keywordvalue for <b>Title</b> (assigned for this server control): home; page; company</p> <p>Metadata values for a content item's <b>Title</b> metadata definition field: software; ektron; company</p> <p>If KeyWordValueMatchAll = true, content does not appear on metadata list because some items do not match</p> <p>If KeyWordValueMatchAll = false, content item appears on metadata list because one item (company) matches</p>	Boolean
KeyWordValueSeparator	Enter the character used to separate the list of keyword values. An example is a semicolon(;).	String
Language	Set a language for viewing the MetaDataList. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer

Property	Value	Data Type
LinkTarget	<p>Defines the way a link acts when a link is clicked. Choices are:</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned BASE target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>	String
MarkupLanguage	<p>Identify the template markup file that controls the display of the metadata list. For example, mymetadatalistmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \siteroot\workarea\customfiles\markup\mymetadatalist.ekml.</p> <p><i>See Also:</i> “<a href="#">Ektron Markup Language</a>” on page 754 and “<a href="#">metadatalist.ekml</a>” on page 791</p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>	String

Property	Value	Data Type
MaxNumber	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, set the <code>EnablePaging</code> property to <b>true</b>.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="564 613 973 840"> <p><a href="#">Bill Lumbergh</a>  <a href="#">Bob Porter</a>  <a href="#">Bob Slydell</a>  <a href="#">Milton Waddams</a>  <a href="#">[First]</a> <a href="#">[Previous]</a> <a href="#">[Next]</a> <a href="#">[Last]</a></p> </div>	Integer
OrderBy	<p>The order of the list to be returned.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>ID</b> - The content block ID number</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>LastEditorLname</b> - The last editor's last name</li> <li>• <b>LastEditorFname</b> - The last editor's first name</li> </ul>	String
Recursive	Whether to search sub-folders of the identified root folder. The starting folder is identified in the <code>FolderID</code> property.	Boolean
SortOrder	Ascending or Descending	String
SuppressWrapperTags	This property is set to <code>false</code> because Ajax uses <code>&lt;div&gt;</code> tags to rewrite the region around the tag. You <i>cannot</i> change the value to <code>true</code> .	Boolean



Property	Value	Data Type
WrapTag	<p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an inline portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p>	String

## Retrieving the XML Structure of a MetadataList

Retrieving the XML structure of XML content allows for greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a MetadataList server control onto it.
3. Set the KeyWordName and KeyWordValue properties.
4. Drag and drop a Textbox on the Web form.
5. Set the TextMode property to MultiLine.

---

**NOTE** It is also recommended that you set the width of the text box to at least 400px.

---

6. On the code behind page, add the following line.  

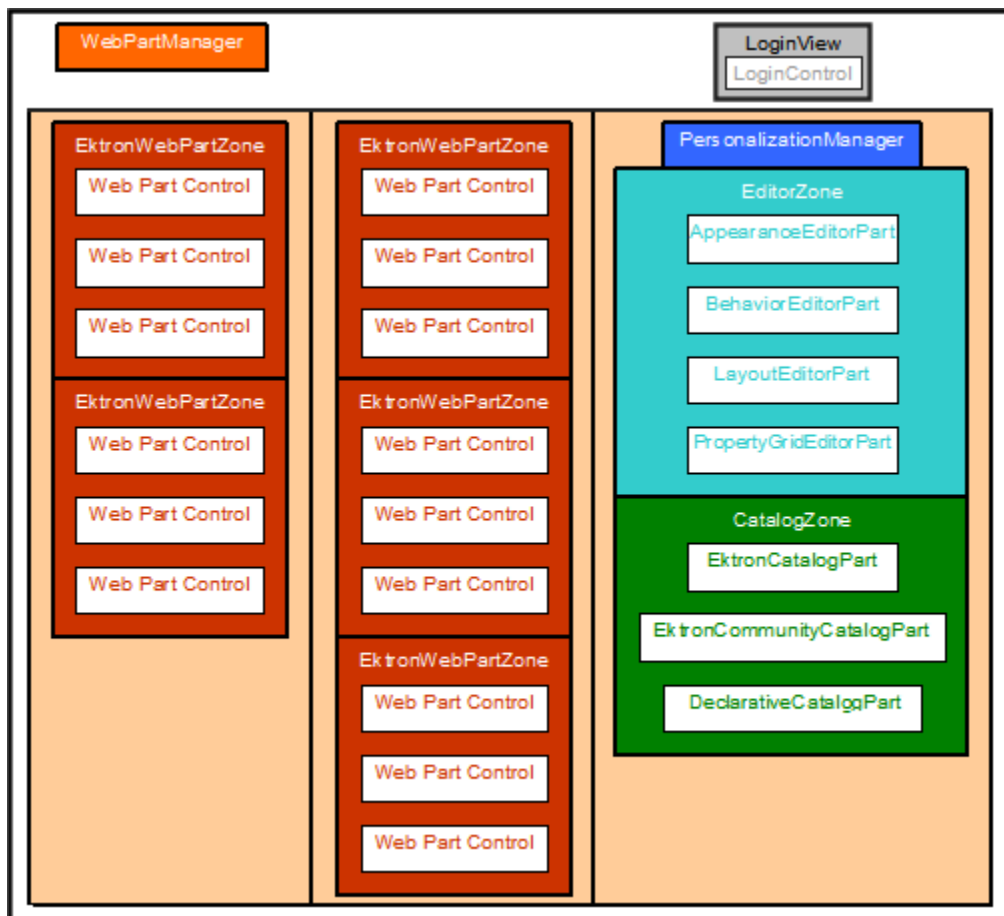
```
Textbox1.Text = Metadata1.XmlDoc.InnerXml
```
7. Build the project.
8. View the Web form in a browser.
9. The XML structure of the MetadataList appears in the textbox.

# Personalization Server Controls

**IMPORTANT!** Understanding how site users and Administrators interact with Personalization helps you create an easier to use personalization Web page. Review the User Manual section “Personalizing a Web Page” and the Administrator Manual section “Personalization Feature”

**IMPORTANT!** The EktronCatalogPart, EktronCommunityCatalogPart, EktronWebPartZone and PersonalizationManager server controls are used in conjunction with Microsoft's WebParts. You should have a good understanding of Microsoft's WebParts when using these server controls. For information on Microsoft's WebParts, see <http://msdn2.microsoft.com/en-us/library/e0s9t4ck.aspx>.

Personalization server controls allow you to create a Web page that site users can personalize when they are logged into the site. Below is a typical layout for a Personalization Web page utilizing Ektron and Microsoft Web Parts.



Ektron CMS400.NET can easily be integrated with Microsoft's WebParts in Visual Studio. Ektron CMS400.NET's server controls are configured for use as WebParts within a EktronWebPartZone in VS. By combining the simplicity of Ektron CMS400.NET and the power of Visual Studio 2005, developers can create pages that users can easily personalize their Ektron CMS400.NET Web site.

Ektron provides four server controls for site personalization:

- **EktronCatalogPart** - when placed inside a Microsoft CatalogZone, it allows site users to choose the type of CMS400.NET control to be added to a WebPartZone. See Also: "[EktronCatalogPart Server Control](#)" on page 279
- **EktronCommunityCatalogPart** - when placed inside a Microsoft CatalogZone, it allows site users to add Discussion Boards (Forums) and Blogs to their Web page. Which Discussion Boards (Forums) and Blogs a site user can add depends on their permissions in Ektron CMS400.NET. See Also: "[EktronCommunityCatalogPart Server Control](#)" on page 282
- **EktronWebPartZone** - provides a location for content to be displayed on a Web page. A site user can move content from one EktronWebPartZone to another when they are logged into the site. The EktronWebPartZone is similar to a Microsoft WebPartZone. The difference is in the way each is formatted. The Microsoft WebPartZone is setup to be a generic WebPartZone. The EktronWebPartZone is designed for use specifically with the CMS400.NET. See Also: "[EktronWebPartZone Server Control](#)" on page 283
- **PersonalizationManager** - provides a control that allows site users to personalize their Web page. When personalizing a Web page, site users can add content, move content from WebPartZone to WebPartZone, and Hide content. See Also: "[PersonalizationManager Server Control](#)" on page 284

## Microsoft WebParts

Since Ektron's Personalization server controls are built off of the Microsoft WebParts platform, you must add the following Microsoft WebParts to your page for the personalization page to work correctly.

- **LoginView** - the LoginView control allows you to display different information to anonymous and logged-in users. The control displays one of two templates: the AnonymousTemplate or the LoggedInTemplate. In the templates, you can add markup and controls that display information appropriate for anonymous users and authenticated users, respectively.
  - **Login** - the Microsoft Login control displays a user interface for user authentication. The Login control contains text boxes for the user name and password and a check box that allows users to indicate whether they want the server to store their identity using ASP.NET membership and automatically be authenticated the next time they visit the site.
  - **Login Status** - the Microsoft LoginStatus control displays a login link for users who are not authenticated and a logout link for users who are authenticated. The login link takes the user to a login page. The logout link resets the current user's identity to be an anonymous user.

- **WebPartManager** - controls all of the Web parts controls on a Web page. Only one WebPartManager is allowed on each page. The WebPartManager needs to appear first on the page before any other Web Part Controls. See Also: <http://msdn2.microsoft.com/en-us/library/ms366728.aspx>
- **EditorZone** - allows users to select content for a control inside a Web part. It also serves as a host for other Editor WebParts:

---

**IMPORTANT!**

While you must add an EditorZone to your Web page, the EditorParts listed below are optional. The EditorZone on a Web page with no other EditorParts allows a site user to choose the content for a control that appears in a WebPartZone.

---

- **AppearanceEditorPart** - enables site users to customize the look of an EktronWebPartZone at run time, such as width, height and title. See Also: <http://msdn2.microsoft.com/en-us/library/ms366716.aspx>
- **BehaviorEditorPart** - enables site users to customize the behavior of an EktronWebPartZone at run time, such as displaying minimize, close and edit buttons. By default, when an EditorZone is added to a page, EktronWebPartZones display minimize, close and edit buttons. Which buttons appear depend on a user's privileges in Ektron CMS400.NET. See Also: <http://msdn2.microsoft.com/en-us/library/ms366717.aspx>
- **LayoutEditorPart** - enables site users to customize the layout properties of an EktronWebPartZone at run time, such as selecting between a minimized and normal state. See Also: <http://msdn2.microsoft.com/en-us/library/ms366711.aspx>
- **PropertyGridEditorPart** - enables site user to edit the custom properties associated with a control in an EktronWebPartZones at run time. For example, if a Collection control is added to an EktronWebPartZone, a site user can set the DisplayXslt property for the control.

For additional information on the EditorZone server control, see <http://msdn2.microsoft.com/en-us/library/ms366516.aspx>

- **CatalogZone** - hosts catalog parts that enable site users to choose the types of controls or content displayed in EktronWebPartZones. The following is a list of catalog parts that are used with a CatalogZone:

---

**IMPORTANT!**

You must add a CatalogZone and at least one catalog part to your Web page. If you choose, you can add multiple catalog parts.

---

- **EktronCatalogPart** - allows users to choose the type of CMS400.NET control that is added to a WebPartZone. See Also: "[EktronCatalogPart Server Control](#)" on page 279
- **EktronCommunityCatalogPart** - allows site users to choose from any available Discussion Boards and Blogs and add them to a WebPartZone. See Also: "[EktronCommunityCatalogPart Server Control](#)" on page 282

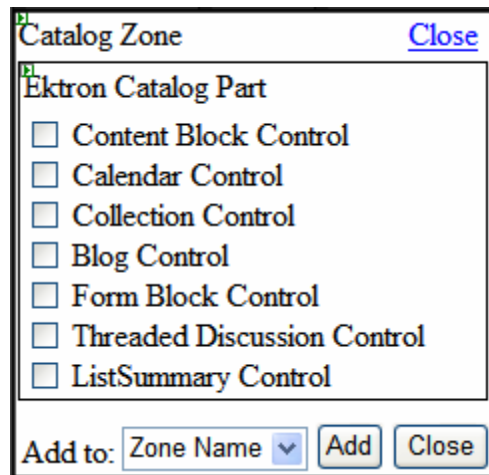
- **DeclarativeCatalogPart** - allows you to create a predefined list of content that site users can select from. See Also: <http://msdn2.microsoft.com/en-us/library/ms227985.aspx>

For additional information on the CatalogZone server control, see <http://msdn2.microsoft.com/en-us/library/ms227557.aspx>

## EktronCatalogPart Server Control

When placed inside a Microsoft CatalogZone, the EktronCatalogPart allows users to choose the type of CMS400.NET control that is added to a WebPartZone. It also allows a developer to predefine content for each control. This way, a site user does not see an empty WebPartZone after adding a control. If content is not defined for a control, the site user can then edit the EktronWebPartZone in which the control is placed.

Below is an example of what the EktronCatalogPart looks like when it is placed in Microsoft CatalogZone.

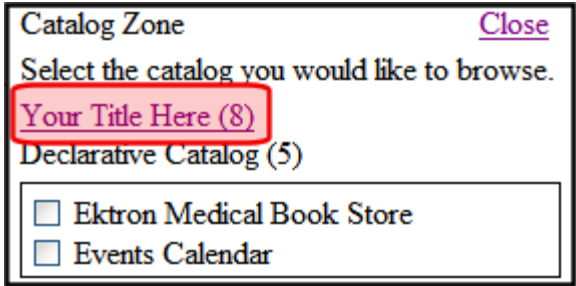


## EktronCatalogPart Server Control Properties


The EktronCatalogPart server control properties are described in this table.

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

**NOTE** The properties below are broken down by category. The category title is in bold.

Property	Description	Data Type
<b>Ektron</b>		
Title	<p>The descriptive name of the Ektron Catalog. The title appears as clickable link in the catalog when the Ektron Catalog is not shown.</p> 	String
<b>Ektron Blog Control</b>		
BlogID	The ID of the blog in CMS400.NET. For example: 41. If you don't know the ID of the blog, use the CMS Explorer to browse to it. See A/so: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site"</a> on page 15	Integer
EnableBlogControl	Allows a Blog Control to appear in the catalog.	Boolean
<b>Ektron Calendar Control</b>		
CalendarDynamicParameter	To make this calendar dynamic, select <b>calendar_id</b> . When you do, this server control uses the calendar passed as a URL parameter.	String
DefaultCalendarID	The ID number of the calendar that appears where you inserted this server control. If you don't know the ID number of the calendar, use the CMS Explorer to browse to it. See A/so: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site"</a> on page 15	Integer
EnableCalendarControl	Allows a Calendar Control to appear in the catalog.	Boolean
<b>EktronCollectionControl</b>		

Property	Description	Data Type
CollectionDynamicParameter	To make this collection dynamic, select <b>id</b> . When you do, this server control uses the collection passed as a URL parameter.	String
DefaultCollectionID	The ID of a collection that appears where you inserted this server control if no other collection is identified, or is not available.  If you don't know the ID number of the collection, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer
EnableCollectionControl	Allows a Collection to appear in the catalog.	Boolean
<b>Ektron Content Block</b>		
ContentBlockDynamic	Select <b>id</b> . When you do, this server control uses the content block passed as a URL parameter.	String
DefaultContentID	The ID of a content block that appears where you insert this server control.  If you don't know the ID number of the content block, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer
EnableContentBlockControl	Allows a Content Block Control to appear in the catalog.	Boolean
<b>Ektron FormBlock Control</b>		
DefaultFormID	The ID of a FormBlock that appears where you inserted this server control if no other form block is identified.  If you don't know the ID number of the form block, use the CMS Explorer to browse to it. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.</i>	Integer
EnableFormBlockControl	Allows a FormBlock Control to appear in the catalog.	Boolean
FormBlockDynamicParameter	To make this form block dynamic, select <b>ekfrm</b> . When you do, this server control uses the form block passed as a URL parameter.	String

Property	Description	Data Type
<b>Ektron ListSummary</b>		
EnableListSummaryControl	Allows a ListSummary Control to appear in the catalog.	Boolean
FolderID	The folder that contains the content blocks which appear in the list summary. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer
<b>Ektron Discussion Board Control</b>		
BoardID	The ID of the discussion board. If you don't know the ID, click the Ellipses button (  ), then sign in, browse to and select the discussion board. See <i>Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</i>	Integer
EnableThreadedDiscussionsControl	Allows a Discussion Board Control to appear in the catalog.	Boolean

## EktronCommunityCatalogPart Server Control

The EktronCommunityCatalogPart Server Control gets a list of Discussion Board (Forums) and Blogs from your site. Then, it allows site users to choose from any available Discussion Boards and Blogs and add them to a WebPartZone. Availability depends on a site users permissions in CMS400.NET.

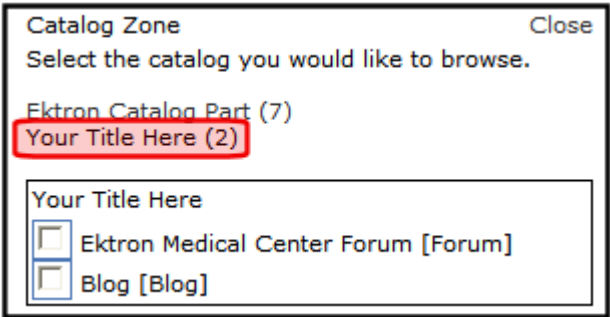
This server control differs from the EktronCatalogPart server control because a user does not have to edit a Web part zone and choose a Discussion Board or Blog. The EktronCommunityCatalogPart server control lists all of a site user's available Discussion Boards and Blogs in the catalog.

### EktronCommunityCatalogPart Server Control Properties

The EktronCommunityCatalogPart server control properties are described in this table.

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.



Property	Description	Data Type
EnableBlogs	Decide if site users are allowed to add blogs.	Boolean
EnableForums	Decide if users are allowed to add Discussion Boards (Forums).	Boolean
Title	<p>The descriptive name of the Ektron Community Catalog. The title appears as clickable link in the catalog when the Ektron Catalog is not shown. The default title is <b>Ektron Community Catalog Part</b>.</p> 	String

## EktronWebPartZone Server Control

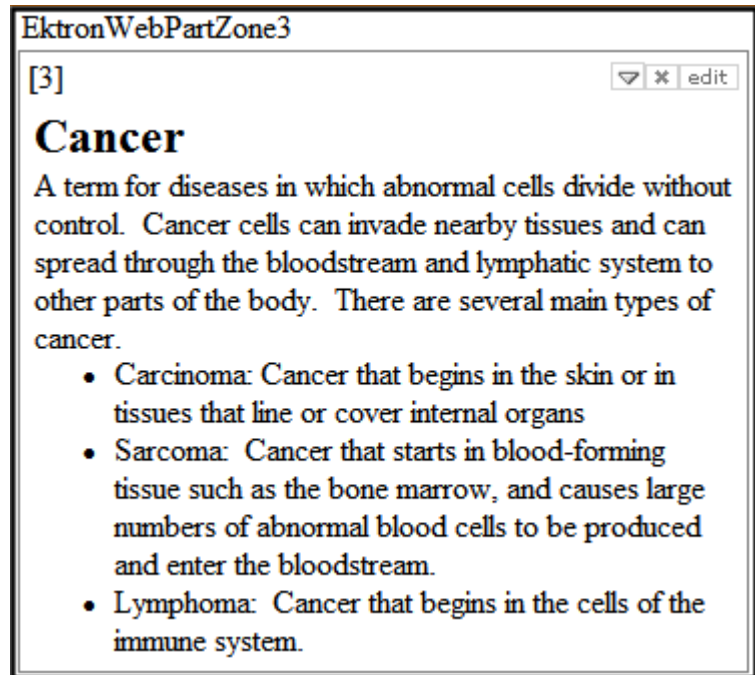
Similar to a Microsoft WebPartZone, the EktronWebPartZone server control provides a location for content to be displayed on a Web page. The difference between the two is formatting. The EktronWebPartZone is designed to work specifically with Ektron Server Controls, while Microsoft WebPartZones are more generic.

EktronWebPartZones are filled by either adding a control during design time or by editing a Web page and using a catalog to add controls. The following list of Ektron Server Controls work with EktronWebPartZones:

- Content Block Control
- Calendar Control
- Collection Control
- Blog Control
- Form Block Control
- Forum Control
- ListSummaryControl

By displaying content in a EktronWebPartZone, site users can add, edit, hide, and move content on their page.

Below is an example of content in an EktronWebPartZone.



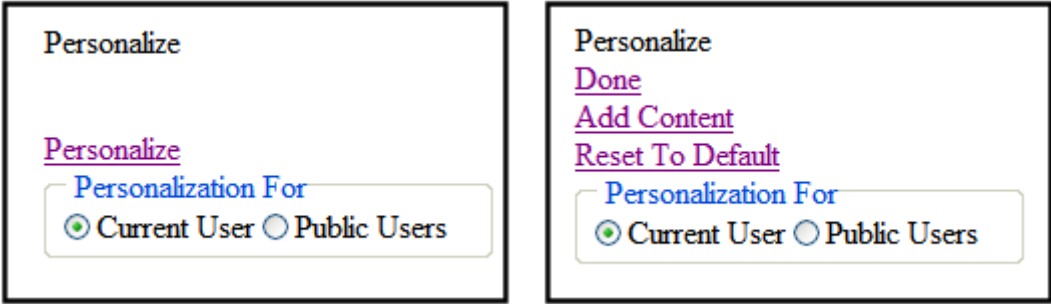
## EktronWebPartZone Server Control Properties

There are no Ektron-specific properties assigned to the EktronWebPartZone server control. The properties associated with the control are the same as the Microsoft WebPartZone. For descriptions of these properties in Visual Studio, click the property and press F1 on your keyboard. This launches context-sensitive Visual Studio Help.

## PersonalizationManager Server Control

The PersonalizationManager server control allows site users to personalize, add, edit and hide content on a Web page. The level of privileges assigned to a site user in CMS400.NET determines what personalization functionality is available when the control is active. The Personalization manager only appears when a user is logged in to the site. See *Also: The Administrator Manual* section "Personalization Feature"

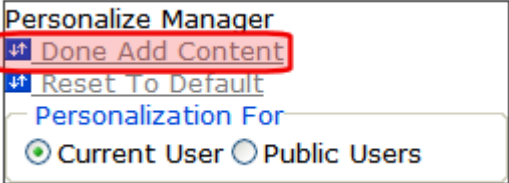
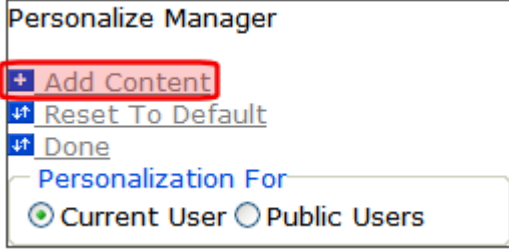
Below are examples of the PersonalizationManager server control on a Web page.

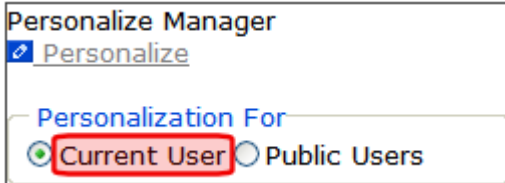
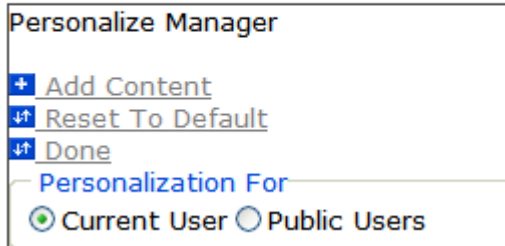
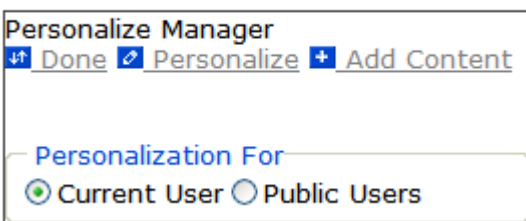



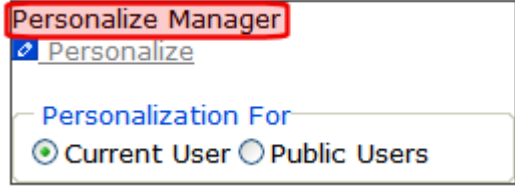
### PersonalizationManager Server Control Properties

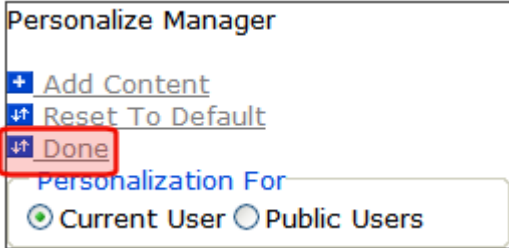
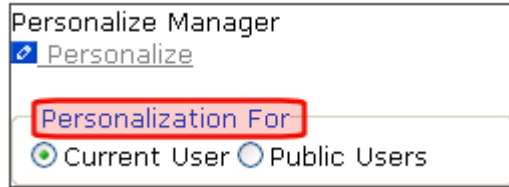


The PersonalizationManager server control properties are described in this table.



**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type
AddContentExitText	Sets the text for the Add Content Exit link. The Add Content Exit link is highlighted below in red. The default is <b>Done Add Content</b> . 	String
AddContentText	Sets the text for the Add Content link. The Add Content link is highlighted below in red. The default is <b>Add Content</b> . 	String

Property	Description	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>	Boolean
CurrentUserButtonText	<p>Sets the text for the Current User radio button. The Current User text is highlighted below in red. The default is <b>Current User</b>.</p> 	String
Display	<p>Sets the display format to vertical or horizontal for the Personalization Manager. The default is <b>Vertical</b>.</p> <p>Vertical</p>  <p>Horizontal</p> 	String

Property	Description	Data Type
HeaderBackColor	<p>Sets the background color of the personalization manager header. The background color is shown in red below. The default is <b>White</b>.</p> 	String
HeaderText	<p>The text that appears in the header. The header text is highlighted below in red. The default is <b>Personalize Manager</b>.</p> 	String
Hide	<p>Used to hide a control in design time and run time.  <b>True</b> = Hide Business Rule  <b>False</b> = Show Business Rule</p>	Boolean
Language	<p>Set a language for Personalization Manager. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p>	Integer

Property	Description	Data Type
PersonalizeExitText	<p>Sets the text for the Exit link. The Exit link is highlighted below in red. The default is <b>Done</b>.</p> 	String
PersonalizeScopeText	<p>Sets the text for the Scope. The Scope text is highlighted below in red. The default is <b>Personalization For</b>.</p> 	String
PersonalizeStartText	<p>Sets the text for the Start link. The Start text link is highlighted below in red. The default is <b>Personalize</b>.</p> 	String
PublicUserButtonText	<p>Sets the text for the Public User radio button. The Public User text is highlighted below in red. The default is <b>Public User</b>.</p> 	String

Property	Description	Data Type
ResetLinkText	<p>Sets the text for the Reset link text. The Reset link text is highlighted below in red. The default is <b>Reset To Default</b>.</p> 	String
ResetLinkToolTip	<p>Sets the tool tip text for the Reset link. The Reset link tool tip text is below in yellow. The default is <b>Reset the current user's personalization data for the page</b>.</p> 	String
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags.</p> <p><b>False</b> - Allow wrap tags.</p>	Boolean
WrapTag	<p>Allows a developer to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p>	String

## Building a Personalization Web Page

This section explains the steps necessary to build a Personalization Web page. When building a Personalization page, the following server controls are used:

- **LoginView** - See Also: <http://msdn2.microsoft.com/en-us/library/ms178329.aspx>
- **Login** - See Also: <http://msdn2.microsoft.com/en-us/library/ms178329.aspx>
- **Login Status** - See Also: <http://msdn2.microsoft.com/en-us/library/ms178329.aspx>
- **EktronCatalogPart** - See Also: "EktronCatalogPart Server Control" on page 279
- **EktronCommunityCatalogPart** - See Also: "EktronCommunityCatalogPart Server Control" on page 282
- **EktronWebPartZone** - See Also: "EktronWebPartZone Server Control" on page 283
- **PersonalizationManager** - See Also: "PersonalizationManager Server Control" on page 284
- **EditorZone** - See Also: <http://msdn2.microsoft.com/en-us/library/ms366516.aspx>
- **CatalogZone** - See Also: <http://msdn2.microsoft.com/en-us/library/ms227557.aspx>
- **WebPartManager** - See Also: <http://msdn2.microsoft.com/en-us/library/ms366728.aspx>

---

**IMPORTANT!**

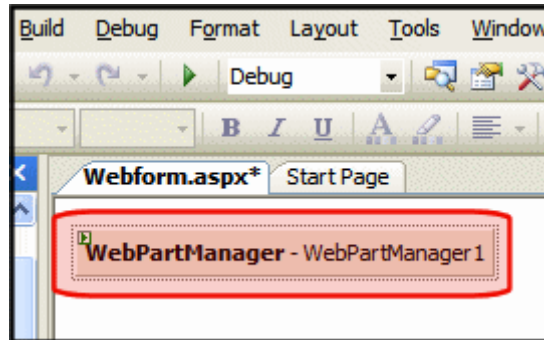
For a personalization Web page to work, a site user needs to be logged in to the site.

---

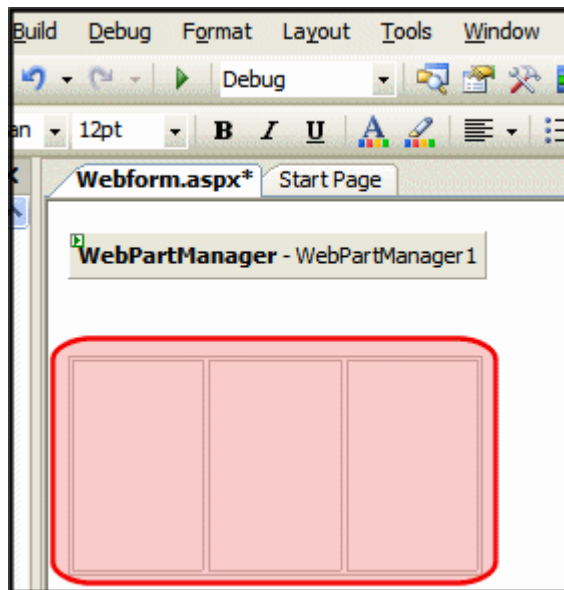
1. Create a new Web form for your site in Visual Studio 2005.
2. Drag a LoginView control from the Login tab in the toolbox.
3. Drag a Microsoft Login control into the LoginView.
4. Click the top of the LoginView control and then click the small black arrow.
5. Change the View to **LoggedInTemplate**.
6. Drag a LoginStatus control into the LoginView.



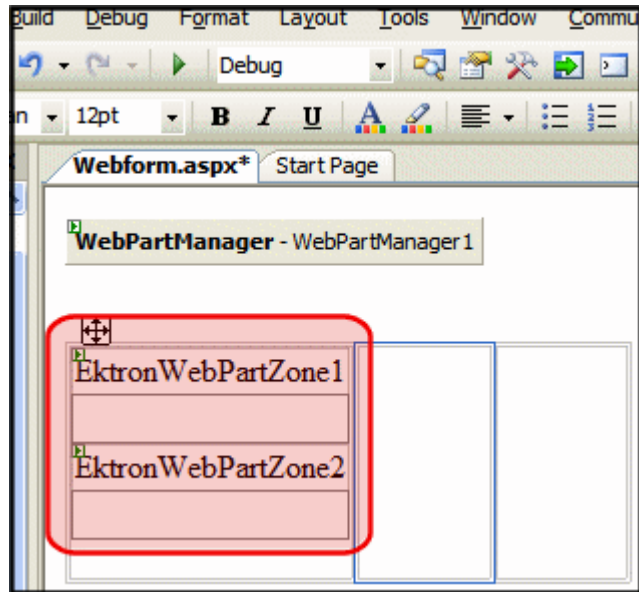
7. Add a WebPartsManager server control to the page. The WebPartsManager needs to be located first on the page, before any other Web parts server control.



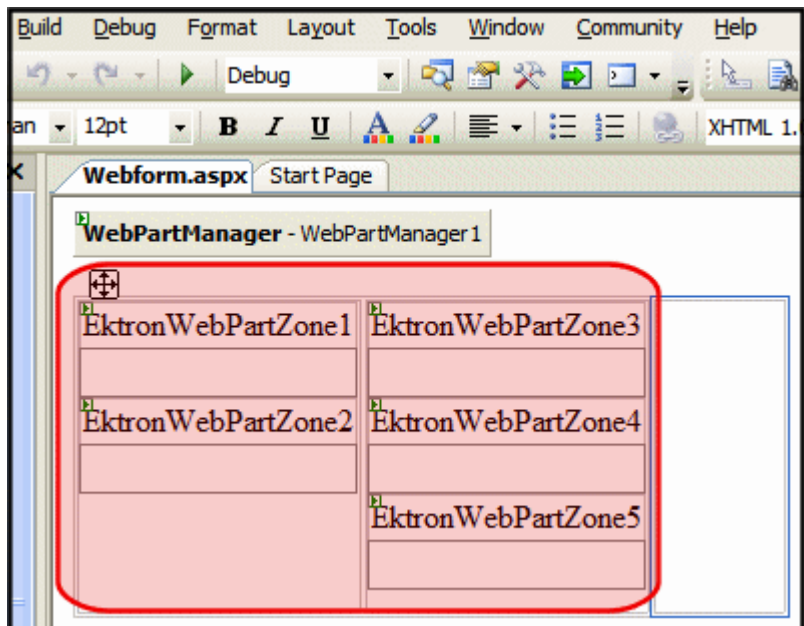
8. Add an HTML Table with one row and three columns.



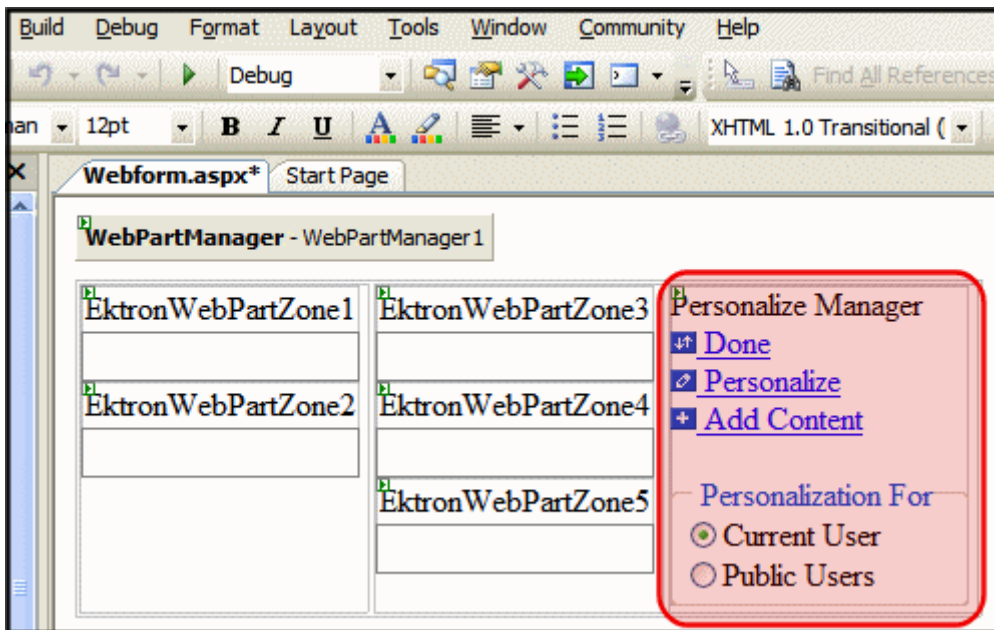
9. In the first column, add two EktronWebPartZones



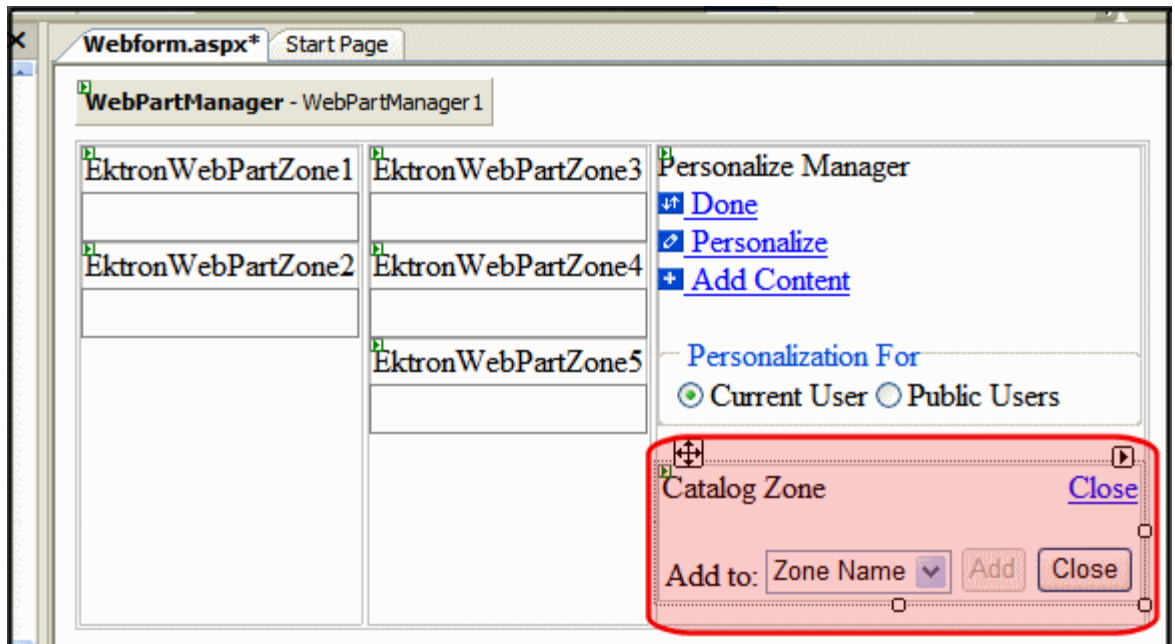
10. Add three EktronWebPartZones server controls to the second column of the table.



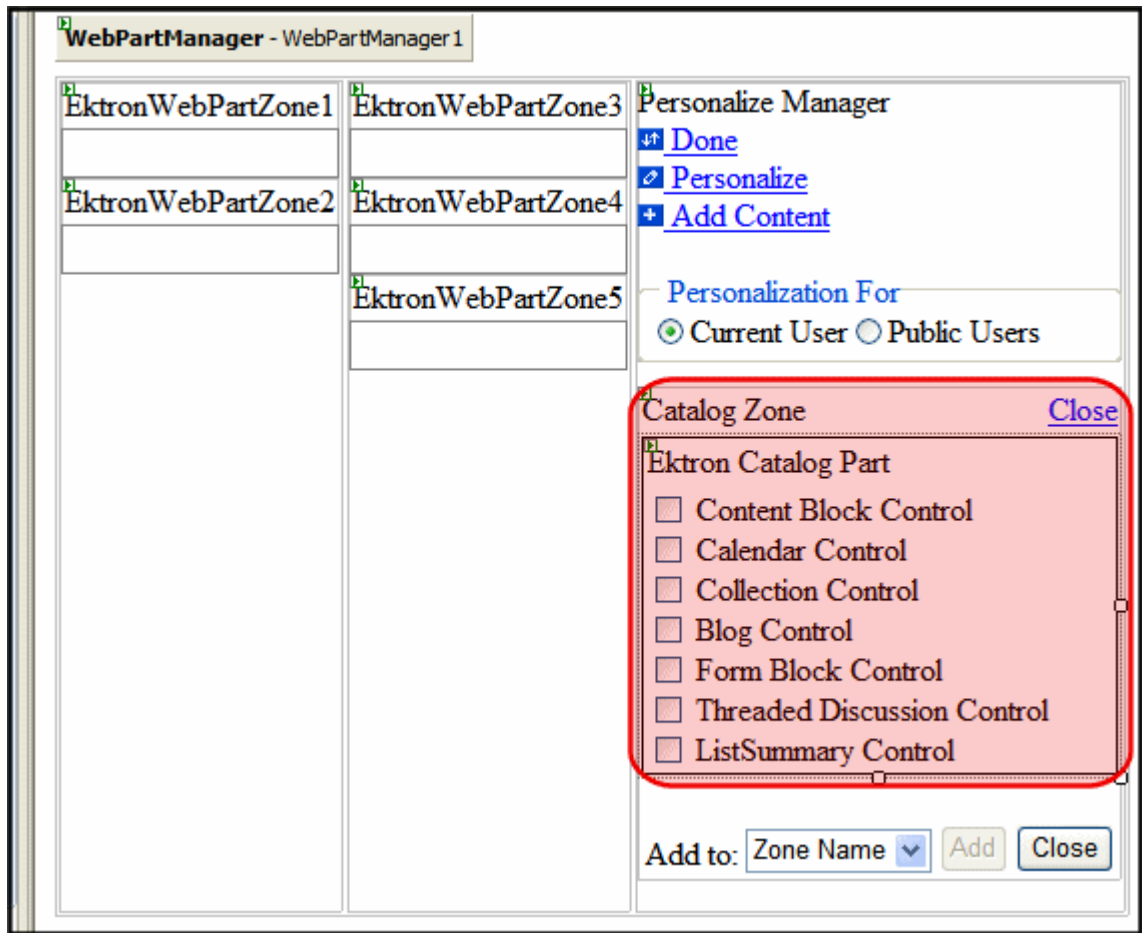
11. In the third column, add a PersonalizationManager server control.



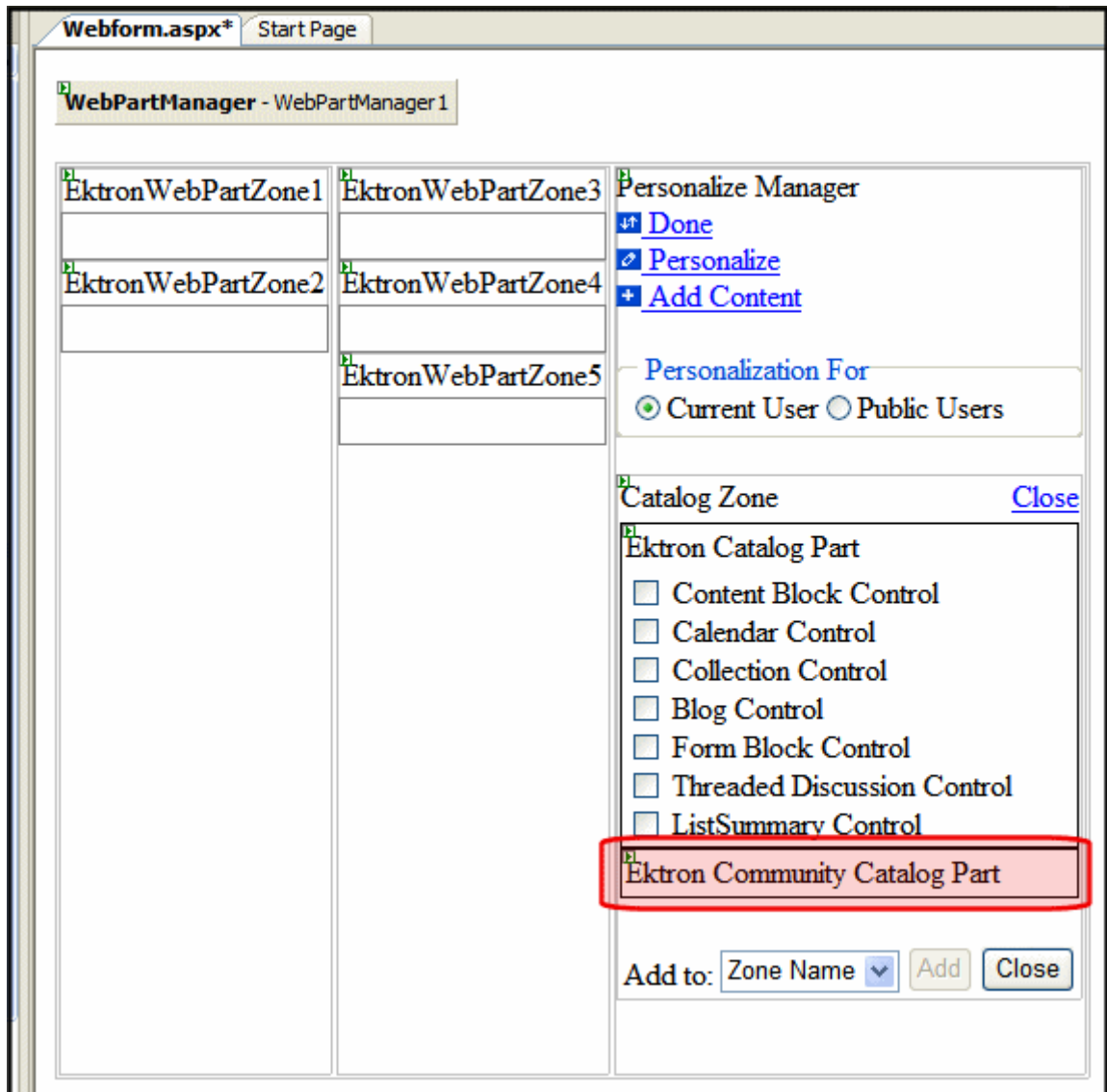
12. Add a CatalogZone server control below the PersonalizationManager server control.



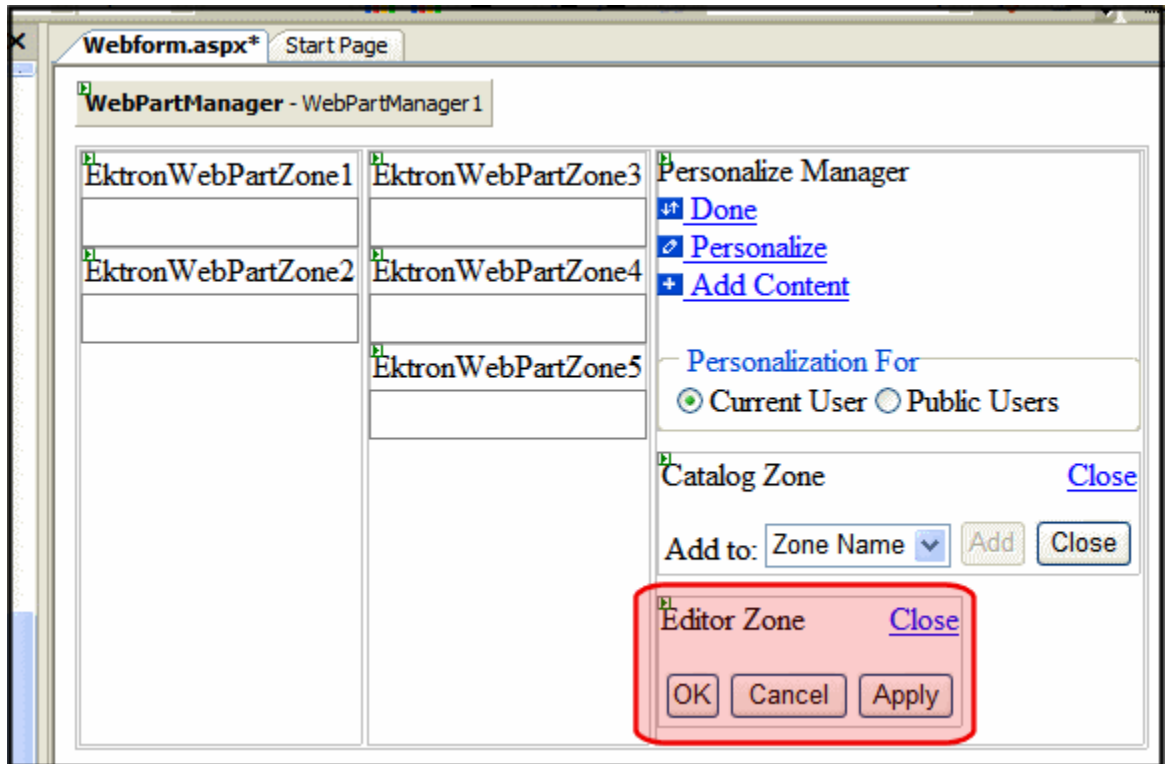
13. Drag and drop an EktronCatalogPart server control on the CatalogZone server control.



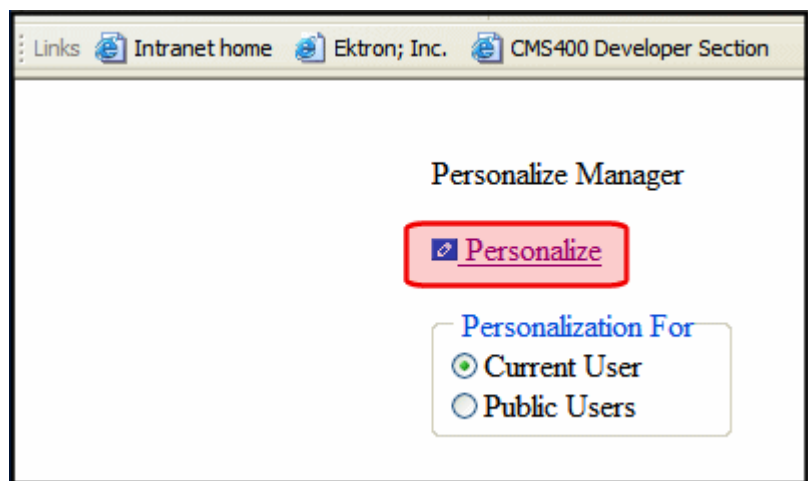
14. Add an EktronCommunityCatalogPart below the EktronCatalogPart.



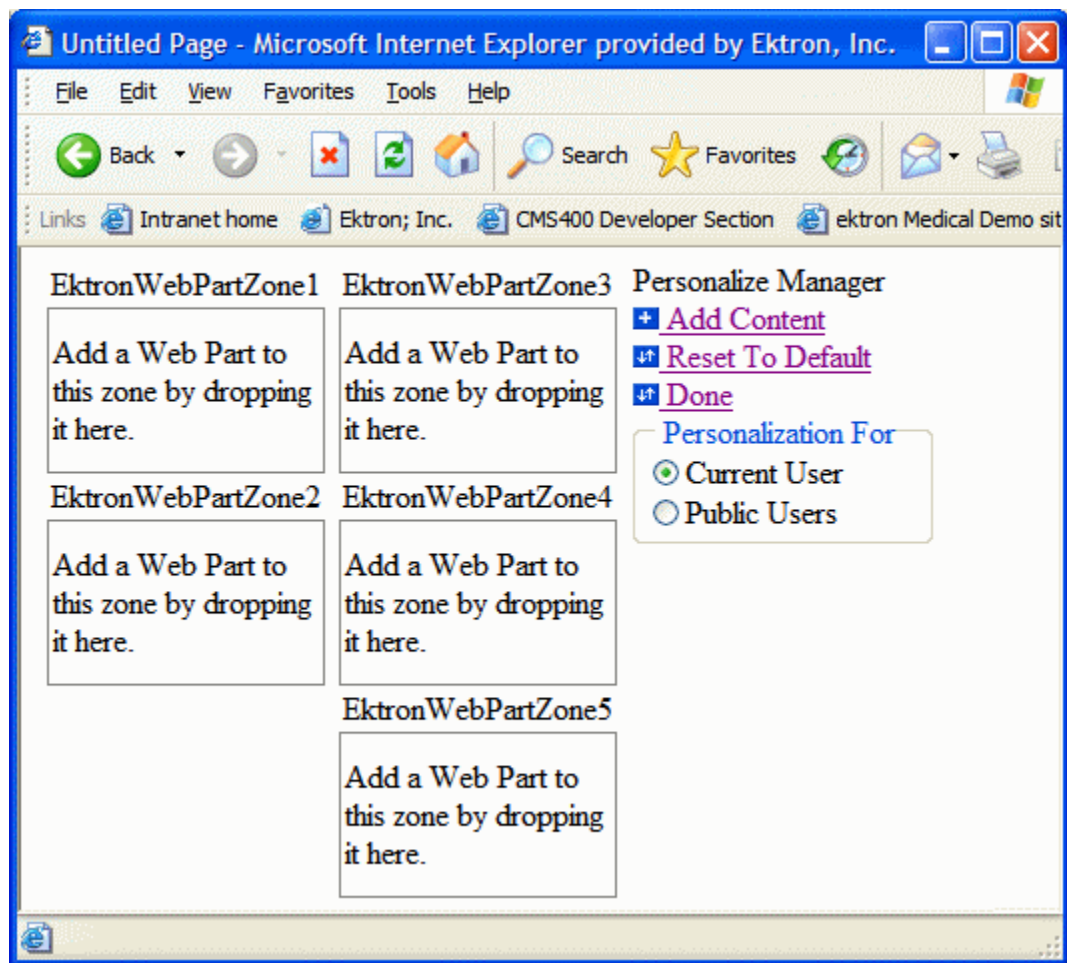
15. Add a EditorZone server control below the CatalogZone server control.



16. Build the page.
17. Congratulations! You now have a basic personalization Web page. To test this page, log in to the site and navigate to the Web page.
18. Click the Personalize link to start personalizing the page.



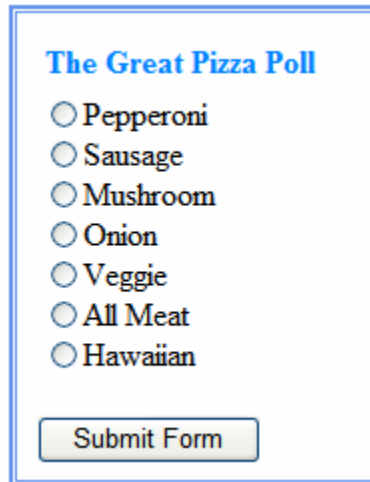
19. Once the Personalize link is clicked, the page shows the EktronWebPartZones and Personalization Manager selections



Using the Personalization feature to add, move, hide, and remove content is explained in the User Manual section "Personalizing a Web Page". Review that section to learn how site users work with a Personalization Web page.

# Poll Server Control

The Poll server control displays a poll or survey created from an Ektron CMS400.NET form on a Web page. When added to a template and visited, the poll might look like this. You can change a poll's appearance by modifying its properties.



**The Great Pizza Poll**

☐ Pepperoni

☐ Sausage

☐ Mushroom

☐ Onion

☐ Veggie

☐ All Meat

☐ Hawaiian

While you can use a FormBlock server control to add a poll or survey, Ektron recommends using a poll server control, because it provides greater flexibility on the poll's appearance.

Typically, developers or site administrators want a poll or survey to be a small part of a Web page not the main content. By using the EnableAjax property you can display the results in the same area as the poll or survey without disturbing the surrounding content.

For a detailed description of these properties, click on them or see ["Poll Server Control Properties" on page 298](#).

This subsection contains the following topics:

- ["Poll Server Control Properties" on page 298](#)
- ["Automatic versus Manual Generation of Form Tags" on page 300](#)

## Poll Server Control Properties

The following table explains the properties of the poll server control.

<b>NOTE</b>	The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.
-------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



Property	Value	Data Type
(Poll ID)	The ID of the poll that appears where you inserted this server control.  If you don't know the ID number of the form block, use the CMS Explorer to browse to it. <i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.</a>	Integer
AddValidation	Set to <b>True</b> to add Validation to your main runat=server form.	Boolean
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. <i>See Also:</i> <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15.</a>	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). <i>See Also:</i> <a href="#">"Caching with Server Controls" on page 42.</a>	Integer
DynamicParameter	To make this form block dynamic, select <b>id</b> . When you do, this server control uses the form block passed as a URL parameter.	String
EnableAjax	Displays the poll or results, using an iFrame, in the area of the page that contains the poll without disturbing its surrounding contents. The default is true.  <b>True</b> = Polls and results are shown in an iFrame without modification or a refresh of surrounding contents.  <b>False</b> = The poll and the results replace the Web page's content.	Boolean
Fields	Displays a list of fields that are defined in the form. These fields are read only. This is an excellent way of displaying the field names used on the form. With this list of names, you can create events using the fields without having to enter the Workarea to see the names.	

Property	Value	Data Type
Hide	Used to hide a form block in design time and run time. <b>True</b> = Hide form block <b>False</b> = Show form block	Boolean
Include Tags	Determines if tags are generated automatically of manually. See Also: " <a href="#">Automatic versus Manual Generation of Form Tags</a> " on page 300	Boolean
Language	Set a language for viewing form content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is <b>False</b> . <b>True</b> - Suppress wrap tags. <b>False</b> - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code. <b>Custom</b> - Allows you to use a custom tag.	String

## Automatic versus Manual Generation of Form Tags

When using the poll server control's Include tags property, you have two options.

Option	Recommended if you are developing	How to Enable
Let Ektron CMS400.NET generate form tags automatically	A pure script .aspx page	Set Include Tags property to <b>True</b> .

Option	Recommended if you are developing	How to Enable
Modify HTML form tags in the .aspx file	An .aspx page and associated code-behind web form	<p>Set Include Tags property to <b>False</b>.</p> <p>Here is the default .NET generated form tag:</p> <pre>&lt;form id="Form1" method="post" runat="server"&gt;</pre> <p>Modify the form tag as indicated in red:</p> <pre>&lt;form id="Form1" method="post" runat="server" OnSubmit="EkFmValidate(this);"&gt;</pre>

# RssAggregator Server Control

An RSS Aggregator or reader takes an RSS feed and displays the information in a readable format. The RssAggregator server control lets you do the same by processing and displaying an RSS feed from any Web site. This allows you to create an information Web page for news, stories, images, lists of music, or for whatever purpose someone creates an RSS feed.

You can create a multi-level information Web page by using multiple RssAggregator server controls on the same page. For example, you could create a world news Web page by adding RSS feeds from the NY Times, The BBC, and AFP (Agence France-Presse).

The process of adding an RSS Aggregator to your Web site is simple.

1. Choose a Web form where the RSS Aggregator will reside
2. Drag and drop the RssAggregator server control on that Web form
3. Set the URL property to point at the RSS feed. For example, `http://msdn.example.com/rss.xml`
4. Save the Web form

When a visitor to your site views the Web form in a browser, he sees the RSS feed displayed properly. When he refreshes the Web page, any information that the RSS feed provider has updated is now reflected. See Also: ["Using the RssAggregator Server Control" on page 304](#)

By creating a custom XSLT, you can change the way an RSS feed is viewed by your site visitors.

This subsection contains the following topics:

- ["RSSAggregator Server Control Properties" on page 302](#)
- ["Using the RssAggregator Server Control" on page 304](#)

## RSSAggregator Server Control Properties

The RSSAggregator server control properties are described in this table.

---

**NOTE**

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

---

Property	Value	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See <a href="#">Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>	String
DisplayXslt	Determines how the information on the page is displayed <b>None</b> -databind only <b>ecmNavigation</b> - lists the title of every RSS feed item See <a href="#">Also: "Example of ecmNavigation Display" on page 101</a> <b>ecmTeaser</b> - lists a title and a description of every RSS feed item See <a href="#">Also: "Example of ecmTeaser Display" on page 102</a> <b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page	String
Hide	Used to hide an RSS Aggregator server control in design time and run time. <b>True</b> = Hide Metadata server control <b>False</b> = Show Metadata server control	boolean
Language	Set a language for viewing the RssAggregator. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
LinkTarget	Defines the way a link acts when a link is clicked. <b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named. <b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target. <b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like " <b>_self</b> " if the document has no parent. <b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like " <b>_self</b> " if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.	String

Property	Value	Data Type
MaxResults	The Maximum number of items from an RSS feed that are returned (0=unlimited).	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is <b>False</b> . <b>True</b> - Suppress wrap tags. <b>False</b> - Allow wrap tags.	Boolean
URL	The RSS feed path for the server control. For example: <code>http://msdn.example.com/rss.xml</code>	String
WrapTag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an inline portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code. <b>Custom</b> - Allows you to use a custom tag.	String

## Using the RssAggregator Server Control

These steps show how to use the RssAggregator server control.

1. Drag an RssAggregator server control into a template.
2. Set the URL property with the path of the RSS feed you want to display. This example use the following RSS feed:

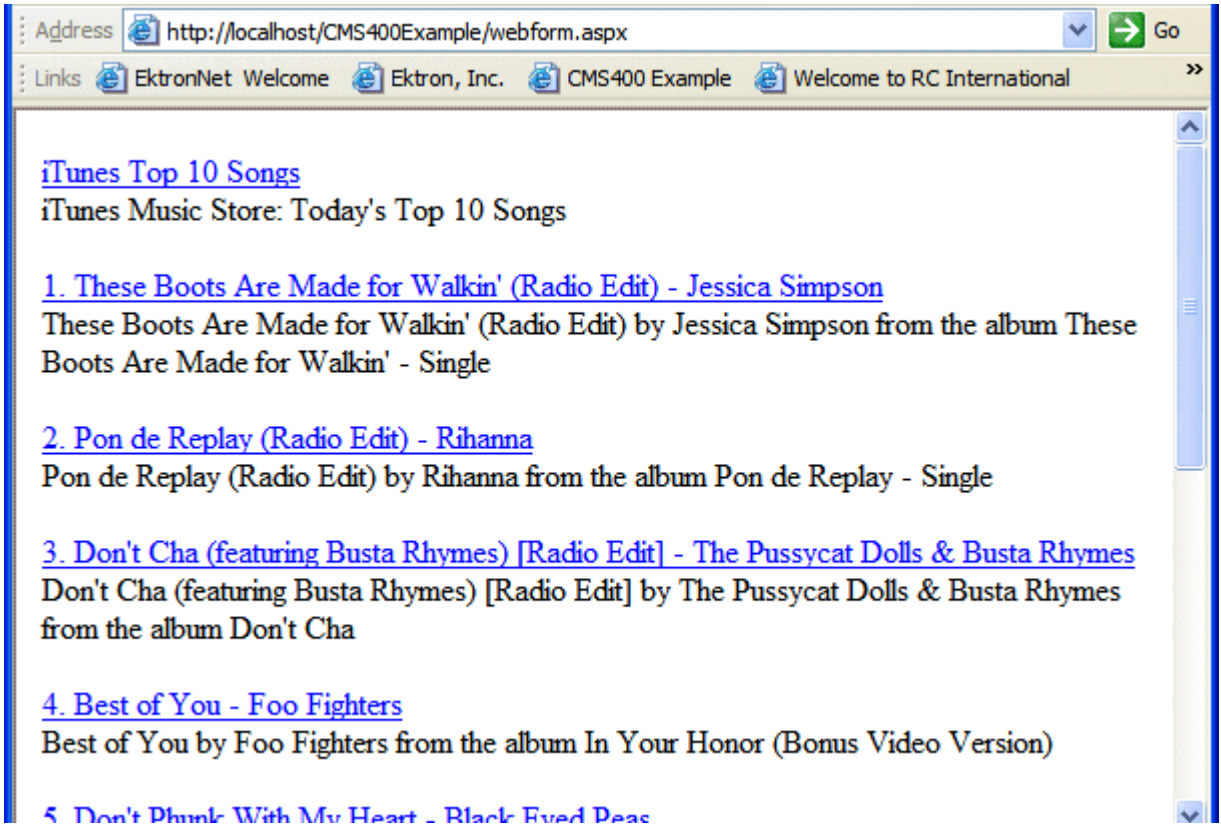
```
http://ax.phobos.apple.com.edgesuite.net/WebObjects/MZStore.woa/wpa/MRSS/topsongs/limit=10/rss.xml
```

This feed displays the top 10 songs according to the iTunes Music Store.

**NOTE** In addition to the URL property, other Ektron properties are available to customize this server control. For more information on these properties, see ["RSSAggregator Server Control Properties" on page 302](#)

3. Build the Web form.

4. Browse to the Web page in your browser.



5. The page appears with an RSS list of information.

# Web Search Server Control

The Web Search server control lets you customize the behavior of the search. You place this control on any Web form from which a site visitor can search your site. For more information about the site visitor experience, see the Ektron CMS400.NET Administrator Manual section "Searching Your Web Site" > "Web Site Search."

## Property Usage Table

Properties in the Search server control generally affect the control in one of three ways:

- **Search Display** - the Search server control's appearance on your site
- **Search Criteria** - criteria used to search the site
- **Search Results Display** - appearance of search results on your site

The following table shows which properties are associated with each category. See Also: "[Property Descriptions](#)" on page 307

Search Display	Search Criteria	Search Results Display
<ul style="list-style-type: none"> <li>• <a href="#">ButtonImgSrc</a></li> <li>• <a href="#">ButtonText</a></li> <li>• <a href="#">EnableAdvanced Link</a></li> <li>• <a href="#">Hide</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">LibraryFieldsetLegend</a></li> <li>• <a href="#">MaxCharacters</a></li> <li>• <a href="#">ShowSearchOptions</a></li> <li>• <a href="#">Text Box Size</a></li> <li>• <a href="#">WrapTag</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">CustomSearch</a></li> <li>• <a href="#">FolderID</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">Recursive</a></li> <li>• <a href="#">SearchFor</a></li> <li>• <a href="#">ShowSearchOptions</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">DisplayXslt</a></li> <li>• <a href="#">EmptyResultMsg</a></li> <li>• <a href="#">Hide</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">LinkTarget</a></li> <li>• <a href="#">OrderBy</a></li> <li>• <a href="#">OrderDirection</a></li> <li>• <a href="#">RemoveTeaserHtml</a></li> <li>• <a href="#">ResultTagId</a></li> <li>• <a href="#">ShowSearchBoxAlways</a></li> </ul> <hr/> <p><b>Note:</b> By default, 10 documents appear on each search result page. To change the number documents per page, use the <code>pageSize</code> element of the <code>&lt;webSearch&gt;</code> <code>&lt;providers&gt;</code> tag in the <i>siteroot/web.config</i> file.</p> <hr/>





## Property Descriptions

The following table describes the Web Search server control properties. See Also: ["Property Usage Table" on page 306](#)

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type	For more information, see
Authenticated	Indicates if you are logged in to the CMS Explorer. If logged in, you can browse to the folder needed for the <b>FolderID</b> property.	String	<a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>
ButtonImgSrc	If you want to display an image on the submit button, enter the server path to it.	String	
ButtonText	The text used for the button if no image source is identified. If an image source is identified, this is alternative text for the button.	String	
CustomSearch	If you want the search to include folders that are not part of <b>Ektron CMS400.NET</b> , enter the folder names here. Separate multiple items with a comma.  You do not need to enter the folder path, but it must reside within the site root folder.	String	<b>Ektron CMS400.NET</b> Administrator Manual > "Searching Your Web Site" > "Using the Integrated Search Feature"

Property	Description	Data Type	For more information, see
DisplayXslt	<p>Determines the display of the search results page.</p> <p><b>None</b> - databind only</p> <p><b>ecmNavigation</b> - lists the title of every content item found by the search</p> <p><b>ecmTeaser</b> - lists the title and the summary of every content item found by the search.</p> <p><b>ecmUnOrderedList</b> - sorts the list in no particular order. Shows the title and content summary.</p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page.</p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>	String	<p><b>ecmNavigation</b> - "Example of ecmNavigation Display" on page 101</p> <p><b>ecmTeaser</b> - "Example of ecmTeaser Display" on page 102</p>
EnableAdvancedLink	<p>Set to <b>true</b> to display a link to the Advanced Search page (see image below).</p> 	Boolean	
FolderID	<p>The folder at which the search begins. The starting folder need not be the root folder.</p> <p>The <b>Recursive</b> property determines if the search examines this folder's subfolders.</p>	String	<p>"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</p> <p>See Also: "Recursive" on page 311</p>
Hide	<p>Used to hide WebSearch Server Control in design time and runtime.</p> <p><b>True</b> = Hide control</p> <p><b>False</b> = Show control</p>	Boolean	

Property	Description	Data Type	For more information, see
Language	<p>If the template on which this server control resides includes a language selection control, and you want to let the site visitor select the language, enter zero (0).</p> <p>Otherwise, click the field, then the ellipsis button (  ) and a popup box appears. Select a language from the list.</p> <p>The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).</p>	Integer	
LinkTarget	<p>Defines how a link on the search results display acts when clicked. The choices are:</p> <p><b>_blank</b> - loads the link in a new blank window. This window is not named.</p> <p><b>_self</b> - loads the link in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - loads the link in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p><b>_top</b> - loads the link in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>	String	

Property	Description	Data Type	For more information, see
MarkupLanguage	<p>Identify the template markup file that controls the display of the search results. For example, mysearchmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \siteroot\workarea\customfiles\markup\mysearchmarkup.ekml.</p> <p>If you enter a valid EkML file, the Displayxslt and Stylesheet property values are ignored.</p>	String	<a href="#">"Ektron Markup Language" on page 754</a> and <a href="#">"websearch.ekml" on page 798</a>
MaxCharacters	<p>The maximum number of characters the <b>Search</b> text box accepts.</p> <p>If you enter a value less than 50, it is a good idea to set the <b>TextBoxSize</b> property to the same number.</p>	Integer	

Property	Description	Data Type	For more information, see
OrderBy	<p>The order of the search results. For example, you want to sort search results by last modified date.</p> <p><b>Title</b> - The content title (alphabetical)</p> <p><b>ID</b> - The content ID number</p> <p><b>Date Created</b> - The date the content was created</p> <p><b>Date Modified</b> - The date the content was most recently modified</p> <p><b>Editor</b> - The user who last edited the content (alphabetical)</p> <p><b>Rank</b> - The rank assigned to the content. <i>See Also:</i> <b>Ektron CMS400.NET</b> Administrator manual section "Searching Your Web Site" &gt; "Display of Search Results" &gt; "Search Result Ranking"</p> <hr/> <p><i>Note: The <b>Order Direction</b> field determines the <i>direction</i> of the search results. For example, if you sort by ID and <b>Order Direction</b> is set to <b>Descending</b>, the results sort by ContentID number with the highest number at the top of the list.</i></p> <hr/>	String	
OrderDirection	<p>The direction in which search results will sort. The default is <b>Ascending</b>.</p> <p><b>Ascending</b> - Alphabetical results from A to Z; numeric values low to high; dates from oldest to most recent</p> <p><b>Descending</b> - Alphabetical results from Z to A; numeric values high to low; dates from most recent to oldest</p>	String	
Recursive	Determines whether to search sub-folders of the starting folder. The starting folder is identified in the <code>FolderID</code> property.	Boolean	<i>See Also:</i> " <b>FolderID</b> " on page 308
RemoveTeaserHTML	Set to true if you want to remove HTML tags from a content summary when it appears in search results.	Boolean	

Property	Description	Data Type	For more information, see
ResultTagId	<p>Lets you designate where search results appear. You can place search criteria in one area of a Web form and the results in another.</p> <p>For example, you have the following tag.</p> <pre>&lt;span id="results"&gt;&lt;/span&gt;</pre> <p>In this case, enter <b>results</b> for the property value.</p>	String	<a href="#">"ResultTagId" on page 313</a>
SearchFor	<p>Choose the type of content to search. See Also: <a href="#">"ShowSearchOptions" on page 312</a></p> <ul style="list-style-type: none"> <li>• All</li> <li>• HTML</li> <li>• Documents</li> <li>• Images</li> <li>• Multimedia</li> <li>• Discussion Forums</li> </ul>	String	
ShowSearchBoxAlways	<p>If set to <b>true</b>, the search box appears on the PostBack screen.</p> <p>If <b>false</b>, the search box does not appear on the PostBack screen.</p> <p>The default is <b>true</b>.</p>	Boolean	
ShowSearchOptions	<p>If set to true, the following options appear above the <b>Search</b> box.</p> <ul style="list-style-type: none"> <li>• Site</li> <li>• Html</li> <li>• Documents</li> <li>• Images</li> <li>• Multimedia</li> </ul>	Boolean	

Property	Description	Data Type	For more information, see
Stylesheet	<p>Specify the location of a style sheet to use for the search results page. Set the location relative to the site root folder. For example:  <code>Workarea\csslib\mytest.css</code>.            Leave blank to use the default style sheet,  <code>\webroot\Workarea\csslib\search.css</code>.</p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Stylesheet property is ignored.</p>	String	<a href="#">"Using a Style Sheet" on page 316</a>
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.</p> <p><b>True</b> - Suppress wrap tags.  <b>False</b> - Allow wrap tags.</p>	Boolean	
Text Box Size	The size of the <b>Search</b> text box for user input. The width is based on the number of characters.	Integer	
WrapTag	<p>Use to specify a server control's tag. The default is <b>Span</b>.</p> <p><b>Span</b> - designates an in-line portion of an HTML document as a span element.  <b>Div</b> - use when you want to apply attributes to a block of code.  <b>Custom</b> - use a custom tag.</p>	String	

## ResultTagId

The `ResultTagId` property lets you designate where search results appear. So, you can place search criteria in one area of a Web form and the results in another.

For example, you have the following tag.

```

```

You would enter **results** for the property value, as shown below.

Recursive	True
RemoveTeaserHtml	True
ResultTagId	results
ShowDate	False
ShowExtendedSearch	True

## Displaying WebSearch Results on a Separate Page

The following example shows how to place a search dialog on one Web form, and display the WebSearch results on a second Web form. You might do this if you want the search dialog to appear in a navigation pane, and the results in a separate Web form. If you are only looking for a code sample and not a walkthrough, see ["Code Example: Displaying WebSearch Results on a Separate Page" on page 316](#)

1. On the search dialog Web form, add a Label, Textbox and Button.
2. Change the Label's text property to **Enter the text to search.**
3. Change the TextBox's ID property to **txtSearchText**.
4. Change the Button's text property to **Search**.
5. Change the Button's ID property to **btnSearch**.
6. When complete, the Web form should look something like this:

7. Add the following code to the code behind page for the Web form. This code redirects the information in the Textbox to the results Web form.

```
Protected Sub btnSearch_Click(ByVal sender As Object, ByVal e As EventArgs) Handles
btnSearch.Click
 Response.Redirect("/[Site Root]/Webform.aspx?searchtext=" + txtSearchText.Text)
End Sub
```

### NOTE

The `Response.Redirect` calls the form where the results are displayed. Change `/siteroot/Webform.aspx` to the location and name of the results form. If the results form is in the same location as the originating form, just use the name of the form. For example, `SearchResultsPage.aspx`.

8. On the results Web form, add a WebSearch server control.
9. If you want to hide the search box dialog box on the results page, set the `ShowSearchBoxAlways` property to **False**.
10. Add `Imports Ektron.Cms.WebSearch.SearchData` to the top of the code behind page for the second Web form.
11. Add the code below to the second Web form's `Page_Load` event.



The first section of code tests whether the Page\_Load is a Callback and Postback. Then, creates and sets the properties of requestData

```
If Not IsCallback AndAlso Not IsPostBack Then
 Dim requestData As SearchRequestData = New SearchRequestData()
 requestData.CurrentPage = 1
 requestData.EnablePaging = True
 requestData.FolderID = 0
 requestData.LanguageID = 1033
 requestData.SearchText = Request.QueryString("searchtext")
```

The next two lines create and load the search results. Note that WebSearch1 is the ID of the WebSearch server control in the example.

```
Dim record_count As Integer = WebSearch1.LoadSearch(requestData)
Dim manager As ClientScriptManager = Page.ClientScript
```

The code below must be added. This code:

- clears the search cookie to prevent any old searches from being performed again
- populates the HTML element identified by the ResultTagId property with the search results (escape quotes in Javascript as \")
- sets the search text box in the server control with the text entered on the previous page

```
If Not manager.IsStartupScriptRegistered("setSearchResults") Then
 Dim newScript as String
 newScript = "/*Clear Search Cookie*/"
 newScript &= "EkSearch.clrCookie();" 'clears the search cookie
 newScript &= "/*Set search results*/"
 newScript &= "document.getElementById('"
 'ID of the results DIV (Don't Change This)
 newScript &= WebSearch1.ResultTagId
 newScript &= "').innerHTML="
 'Insert search results into the page replacing the double-quotes in the search (") so they
 'don't cause interference
 newScript &= WebSearch1.Text.Replace("","\"")
 newScript &= "';"
 newScript &= "/*Set search box to SearchText*/"
 'Sets searchbox text to the search term.
 newScript &= "document.getElementById('ecmBasicKeywords').value = "
 newScript &= "'" & requestData.SearchText & "';"
 'Register Startup Script
 manager.RegisterStartupScript(Me.GetType, "setSearchResults", newScript, True)
End If
End If
```

12. Save all files.
13. In your browser, navigate to the search dialog Web form.
14. Add the search text to the text box and click the search button.

15. A page is displayed with the results. Note, the URL is now the URL of the results page.

### Code Example: Displaying WebSearch Results on a Separate Page

```
Imports Ektron.Cms.WebSearch.SearchData

If Not IsCallback AndAlso Not IsPostBack Then
 Dim requestData As SearchRequestData = New SearchRequestData()
 requestData.CurrentPage = 1
 requestData.EnablePaging = True
 requestData.FolderID = 0
 requestData.LanguageID = 1033
 requestData.SearchText = Request.QueryString("searchtext")

 Dim record_count As Integer = WebSearch1.LoadSearch(requestData)
 Dim manager As ClientScriptManager = Page.ClientScript

 If Not manager.IsStartupScriptRegistered("setSearchResults") Then
 Dim newScript as String
 newScript = "/*Clear Search Cookie*/"
 newScript &= "EkSearch.clrCookie();" 'clears the search cookie
 newScript &= "/*Set search results*/"
 newScript &= "document.getElementById('"
 'ID of the results DIV (Don't Change This)
 newScript &= WebSearch1.ResultTagId
 newScript &= "').innerHTML="""
 'Insert search results into the page replacing the double-quotes in the search (") so
 'they don't cause interference
 newScript &= WebSearch1.Text.Replace("""", "\"\"")
 newScript &= """;"
 newScript &= "/*Set search box to SearchText*/"
 'Sets searchbox text to the search term.
 newScript &= "document.getElementById('ecmBasicKeywords').value = "
 newScript &= "'" & requestData.SearchText & "';"
 'Register Startup Script
 manager.RegisterStartupScript(Me.GetType, "setSearchResults", newScript, True)
 End If
End If
```

### Additional Resources

If you have access to Ektron's Developer samples, refer to <http://<your site>/siteroot/CMS400Developer/default.aspx>. Then, click **Search** to find examples that demonstrate use of the search, such as **Search on Postback**. In addition, view the following Ektron Knowledge Base article.

[http://dev.ektron.com/kb\\_article.aspx?id=7920](http://dev.ektron.com/kb_article.aspx?id=7920)

## Using a Style Sheet

Style Sheets let you customize visual elements such as color, alignment, font, and size. You can create your own style sheet or modify the default one, `siteroot\Workarea\csslib\search.css`.

**Best Practice**

Ektron recommends copying and renaming the default style sheet. Then, enter the new one into the search server control's `Stylesheet` property. This insures you always have a clean file to start with and gives you something to reference if you are not getting the results you expect.

The style sheet must reside underneath the site root.

You can only define one Search style sheet for each Web page. If you place more than one Search server control on a Web page, they share a style sheet.


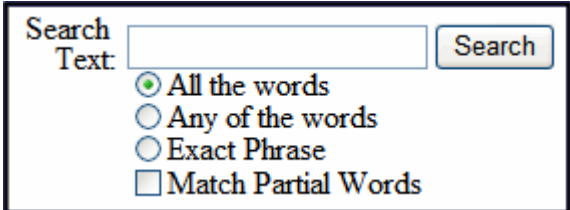
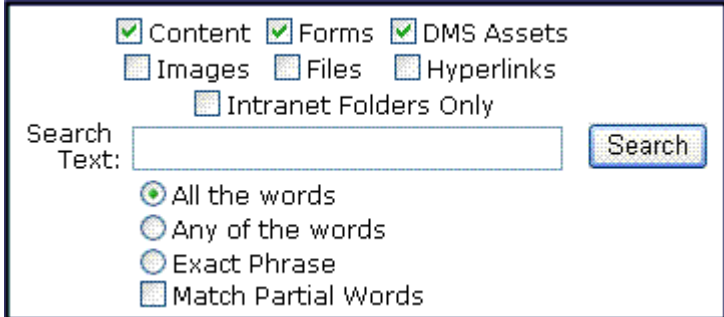
For more information on style sheets, visit <http://www.w3.org/Style/CSS/>.

# Search Server Control

**WARNING!** This server control is deprecated as of the 7.0 Release. For optimal performance, Ektron recommends the Web Search server control. See ["Web Search Server Control"](#) on page 306.

## Introduction

The Search server control displays a text box with a search button on the template. The search is highly customizable and allows you to provide a basic search, a standard search, or advanced search. The different types of searches are explained in ["Using the Advanced Search Properties"](#) on page 339. Below is a visual example of the different searches.

Description	Web Page View
A Basic Search	
A Standard Search	
An Advanced or Mixed Search	

When users perform a search using the Search server control, they can be looking through Ektron CMS400.NET content, DMS assets, forms and library items. To learn more about configuring the Search server control to search

different content types, see ["Using the Advanced Search Properties" on page 339](#).

Searches can also be performed on content that utilizes the Searchable Property metadata type. You can make this search parameter available to users performing a search or hide it. See *Also*: ["ShowExtendedSearch" on page 336](#)

To search XML content (that is, content created with Smart Forms), use the IndexSearch server control. See *Also*: ["Search Server Control vs. IndexSearch Server Control" on page 319](#).

This subsection contains the following topics.

- ["Search Server Control vs. IndexSearch Server Control" on page 319](#)
- ["Search Server Control Properties" on page 319](#)
- ["Property Usage Table" on page 320](#)
- ["Property Descriptions" on page 321](#)
- ["Text Box Examples" on page 333](#)
- ["Button Image Source Examples" on page 334](#)
- ["Search Button Text Examples" on page 335](#)
- ["Display Examples" on page 335](#)
- ["ShowDate Examples" on page 336](#)
- ["ShowExtendedSearch" on page 336](#)
- ["Using the Advanced Search Properties" on page 339](#)
- ["Fieldset Properties" on page 345](#)
- ["Retrieving the XML Structure of a Search" on page 347](#)
- ["Displaying Search Results on a Separate Page" on page 347](#)
- ["Using a Style Sheet" on page 349](#)
- ["Formatting the Search Server Control" on page 350](#)

## Search Server Control vs. IndexSearch Server Control

The Search server control lets users search your Web site for HTML Content (Both active and archived), DMS assets, forms and library items. The IndexSearch server control lets users search XML content on your Web site. For more information on the IndexSearch server control, see ["IndexSearch Server Control" on page 351](#).

## Search Server Control Properties

The following tables explain the properties of the Search server control. The first table shows whether the properties affect the search display, the search criteria or the search results display. The second table contains a list of all the properties,

their descriptions, data type and links to more information. The rest of the tables provide further information on the various Search server control properties.

## Property Usage Table

Properties in the Search server control generally affect the control in one of three ways:

- **Search Display** - the Search server control's appearance on your site
- **Search Criteria** - the criteria the control uses to search the site
- **Search Results Display** - how the search results are displayed on your site

The following table shows which properties are associated with each category.  
See Also: ["Property Descriptions"](#) on page 321

Search Display	Search Criteria	Search Results Display
<ul style="list-style-type: none"> <li>• <a href="#">ButtonImgSrc</a></li> <li>• <a href="#">ButtonText</a></li> <li>• <a href="#">ContentFieldsetLegend</a></li> <li>• <a href="#">Display</a></li> <li>• <a href="#">EnableAdvancedLink</a></li> <li>• <a href="#">EnableArchivedBtn</a></li> <li>• <a href="#">EnableContentBtn</a></li> <li>• <a href="#">EnableContentFieldset</a></li> <li>• <a href="#">EnableDMSAssetsBtn</a></li> <li>• <a href="#">EnableFolderFieldset</a></li> <li>• <a href="#">EnableFormsBtn</a></li> <li>• <a href="#">EnableLibFilesBtn</a></li> <li>• <a href="#">EnableLibHyperlinksBtn</a></li> <li>• <a href="#">EnableLibImageBtn</a></li> <li>• <a href="#">EnableLibraryFieldset</a></li> <li>• <a href="#">FolderFieldsetLegend</a></li> <li>• <a href="#">Hide</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">LibraryFieldsetLegend</a></li> <li>• <a href="#">MaxCharacters</a></li> <li>• <a href="#">ShowExtendedSearch</a></li> <li>• <a href="#">ShowSearchOptions</a></li> <li>• <a href="#">Text Box Size</a></li> <li>• <a href="#">WrapTag</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">EnableArchived</a></li> <li>• <a href="#">EnableContent</a></li> <li>• <a href="#">EnableDMSAssets</a></li> <li>• <a href="#">EnableForms</a></li> <li>• <a href="#">EnableLibFiles</a></li> <li>• <a href="#">EnableLibHyperlinks</a></li> <li>• <a href="#">EnableLibImages</a></li> <li>• <a href="#">FolderID</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">Recursive</a></li> <li>• <a href="#">ShowSearchOptions</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">DisplayXslt</a></li> <li>• <a href="#">EmptyResultMsg</a></li> <li>• <a href="#">Hide</a></li> <li>• <a href="#">IncludeIcons</a></li> <li>• <a href="#">Language</a></li> <li>• <a href="#">LinkTarget</a></li> <li>• <a href="#">MaxResults</a></li> <li>• <a href="#">MaxTeaserLength</a></li> <li>• <a href="#">OrderBy</a></li> <li>• <a href="#">OrderDirection</a></li> <li>• <a href="#">PostResultsAt</a></li> <li>• <a href="#">RemoveTeaserHtml</a></li> <li>• <a href="#">ResultTagId</a></li> <li>• <a href="#">ShowDate</a></li> <li>• <a href="#">ShowSearchBoxAlways</a></li> </ul>

## Property Descriptions

The following table describes the Search server control Properties. See Also: ["Property Usage Table"](#) on page 320

**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type	For more information, see
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to the folder needed for the <b>FolderID</b> property.	String	"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15
ButtonImgSrc	If you want to display an image on the submit button, enter the server path to it.	String	"Button Image Source Examples" on page 334
ButtonText	The text used for the button if no image source is identified. If an image source is identified, this is alternative text for the button.	String	"Search Button Text Examples" on page 335
ContentFieldsetLegend	Text that appears in the fieldset border around the content check boxes. This property is used only when the EnableContentFieldset property is set to True. The default is: Content Items:	String	"Fieldset Properties" on page 345 See Also: "EnableContentFieldset" on page 324
Display	Determines if the search options are in a row ( <b>Horizontal</b> ) or a column ( <b>Vertical</b> ).	String	"Display Examples" on page 335
DisplayXslt	Determines how the information on the page is displayed <b>None</b> -databind only <b>ecmNavigation</b> - lists the title of every content block in the folder <b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary <b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page	String	<b>ecmNavigation</b> - "Example of ecmNavigation Display" on page 101 <b>ecmTeaser</b> - "Example of ecmTeaser Display" on page 102
EmptyResultMsg	The return message displayed if the search returns no hits. The default is: Search returned zero results	String	





Property	Description	Data Type	For more information, see
EnableAdvancedLink	Set to true to display the link that makes the advanced properties available. <b>True</b> = Display Advanced link <b>False</b> = Hide Advanced link	Boolean	<a href="#">"Using the Advanced Search Properties" on page 339</a>
EnableAjax	Set to true to enable Ajax searches. When enabled, the <code>MaxResults</code> property determines the maximum number of results per page, and the <code>ResultTagId</code> property becomes active. <b>True</b> = Enable Ajax Search <b>False</b> = Original HTML Search	Boolean	<a href="#">"Ajax Search" on page 332</a> <i>See Also:</i> <a href="#">"MaxResults" on page 328</a> and <a href="#">"ResultTagId" on page 330</a> .
EnableArchived	Enables users to search archived content. The archived content must be archived as Archived Remain On Site. The default is False. <b>True</b> = Searching archived content is enabled. <b>False</b> = Searching archived content is disabled, unless Include Archived is checked in the Advanced search. For more information on archiving content, see the User Manual section "Scheduling Content to Begin and End" > "Setting Archive Options".	Boolean	<a href="#">"Using the Advanced Search Properties" on page 339</a> <i>See Also:</i> <a href="#">"EnableArchivedBtn" on page 323</a>
EnableArchivedBtn	Provides a check box so users can decide if they want to perform an archive search. The default is False. <b>True</b> = The Include Archive check box is enabled in the advanced search properties. If used with EnableArchived, the check box is checked by default. <b>False</b> = The Include Archive check box is disabled in the advanced search properties.	Boolean	<a href="#">"Using the Advanced Search Properties" on page 339</a>
EnableContent	Enables users to search content. The default is True. <b>True</b> = Searching content is enabled. <b>False</b> = Searching content is disabled, unless Content is checked in the Advanced search.	Boolean	<a href="#">"Using the Advanced Search Properties" on page 339</a> <i>See Also:</i> <a href="#">"EnableContentBtn" on page 324</a>

Property	Description	Data Type	For more information, see
EnableContentBtn	<p>Provides a check box so users can decide if they want to perform a content search. The default is True.</p> <p><b>True</b> = The Content check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default.</p> <p><b>False</b> = The Content check box is disabled in the advanced search properties.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p>
EnableContentFieldset	<p>Creates a border around the group of content items in an advanced search. These content items include Content, Forms and DMS Assets. The default is False.</p> <p><b>True</b> = Show border around content items.</p> <p><b>False</b> = No border around content items.</p>	Boolean	<p>"Fieldset Properties" on page 345</p> <p>See Also: "ContentFieldsetLegend" on page 322</p>
EnableDMSAssets	<p>Enables users to search DMS Assets. The default is True.</p> <p><b>True</b> = Searching DMS Assets is enabled.</p> <p><b>False</b> = Searching DMS Assets is disabled, unless DMS Assets is checked in the Advanced search.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p> <p>See Also: "EnableDMSAssetsBtn" on page 324</p>
EnableDMSAssetsBtn	<p>Provides a check box so users can decide if they want to perform a DMS Asset search. The default is True.</p> <p><b>True</b> = The DMS Asset check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default.</p> <p><b>False</b> = The DMS Asset check box is disabled in the advanced search properties.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p>
EnableFolderFieldset	<p>Creates a border around the group of folders in an advanced search. The default is False.</p> <p><b>True</b> = Show border around folders.</p> <p><b>False</b> = No border around folders.</p>	Boolean	<p>"Fieldset Properties" on page 345</p> <p>See Also: "FolderFieldsetLegend" on page 326</p>

Property	Description	Data Type	For more information, see
EnableForms	Enables users to search Forms. The default is True. <b>True</b> = Searching Forms is enabled. <b>False</b> = Searching Forms is disabled, unless Forms is checked in the Advanced search.	Boolean	"Using the Advanced Search Properties" on page 339 See Also: "EnableFormsBtn" on page 325
EnableFormsBtn	Provides a check box so users can decide if they want to perform a Forms search. The default is True. <b>True</b> = The Forms check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default. <b>False</b> = The Forms check box is disabled in the advanced search properties.	Boolean	"Using the Advanced Search Properties" on page 339
EnableLibFiles	Enables users to search Library Files. The default is True. <b>True</b> = Searching Library Files is enabled. <b>False</b> = Searching Library Files is disabled, unless Files is checked in the Advanced search.	Boolean	"Using the Advanced Search Properties" on page 339 See Also: "EnableLibFilesBtn" on page 325
EnableLibFilesBtn	Provides a check box so users can decide if they want to perform a Library Files search. The default is True. <b>True</b> = The Files check box is enabled in the advanced search properties. If used with EnableLibFiles, the check box is checked by default. <b>False</b> = The Files check box is disabled in the advanced search properties.	Boolean	"Using the Advanced Search Properties" on page 339
EnableLibHyperlinks	Enables users to search Library Hyperlinks. The default is True. <b>True</b> = Searching Library Hyperlinks is enabled. <b>False</b> = Searching Library Hyperlinks is disabled, unless Hyperlinks is checked in the Advanced search.	Boolean	"Using the Advanced Search Properties" on page 339 See Also: "EnableLibHyperlinksBtn" on page 326

Property	Description	Data Type	For more information, see
EnableLibHyperlinksBtn	<p>Provides a check box so users can decide if they want to perform a Library Hyperlinks search. The default is True.</p> <p><b>True</b> = The Hyperlinks check box is enabled in the advanced search properties. If used with EnableLibHyperlinks, the check box is checked by default.</p> <p><b>False</b> = The Hyperlinks check box is disabled in the advanced search properties.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p>
EnableLibImages	<p>Enables users to search Library Images. The default is True.</p> <p><b>True</b> = Searching Library Images is enabled.</p> <p><b>False</b> = Searching Library Images is disabled, unless Images is checked in the Advanced search.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p> <p>See Also: "EnableLibImageBtn" on page 326</p>
EnableLibImageBtn	<p>Provides a check box so users can decide if they want to perform a Library Images search. The default is True.</p> <p><b>True</b> = The Images check box is enabled in the advanced search properties. If used with EnableLibImages, the check box is checked by default.</p> <p><b>False</b> = The Images check box is disabled in the advanced search properties.</p>	Boolean	<p>"Using the Advanced Search Properties" on page 339</p>
EnableLibraryFieldset	<p>Creates a border around the group of library items in an advanced search. These items include: Images, Files and Hyperlinks. The default is False.</p> <p><b>True</b> = Show border around library items.</p> <p><b>False</b> = No border around library items.</p>	Boolean	<p>"Fieldset Properties" on page 345</p> <p>See Also: "LibraryFieldsetLegend" on page 328</p>
FolderFieldsetLegend	<p>Text that appears in the fieldset border around the Folder check boxes. This property is used only when the EnableFolderFieldset property is set to True. The default is:</p> <p>Folder Descriptions:</p>	String	<p>"Fieldset Properties" on page 345</p> <p>See Also: "EnableFolderFieldset" on page 324</p>

Property	Description	Data Type	For more information, see
FolderID	The folder at which the search begins. The starting folder need not be the root folder. The Recursive property determines if the search examines this folder's subfolders.	String	<a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a> See Also: <a href="#">"Recursive" on page 329</a>
Hide	Used to hide Search Server Control in design time and runtime. <b>True</b> = Hide control <b>False</b> = Show control	Boolean	
Includelcons	Choose whether to display icons next to the search result links.  <hr/> <b>Important:</b> This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$ImageIcon] variable is used in an EkML file. <hr/> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">  <a href="#">Ektron Rated Positive</a>            Ektron Inc., an innovator in Web content management software, today announced ...   <a href="#">Ektron to Demonstrate Healthcare</a>            "Healthcare Content Indexing Framework" which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.         </div>	Boolean	
Language	Set a language for viewing the search. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer	

Property	Description	Data Type	For more information, see
LibraryFieldsetLegend	Text that appears in the fieldset border around the Library items check boxes. This property is used only when the EnableLibraryFieldset property is set to True. The default is: <b>Library items:</b>	String	<a href="#">"Fieldset Properties" on page 345</a> See Also: <a href="#">"EnableLibraryFieldset" on page 326</a>
LinkTarget	Defines the way a link acts when a link is clicked. The choices are: <b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named. <b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target. <b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "_self" if the document has no parent. <b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "_self" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.	String	
MaxCharacters	The maximum number of characters the input text box accepts when doing a search.	Integer	
MaxResults	The maximum number of content blocks returned. (0=unlimited). When the EnableAjax property is set to true, the way this property works changes from the maximum amount of results to the maximum results per page. This takes advantage of Ajax's ability to have paging in the searches. If zero is entered, all results appear on the same page.	Integer	<a href="#">"EnableAjax" on page 323</a>
MaxTeaserLength	An optional search result formatting parameter that limits the length of any returned content item's teaser/summary. (0=unlimited)	Integer	

Property	Description	Data Type	For more information, see
OrderBy	<p>The order of the list to be returned. For example, you want to sort search results by last modified date.</p> <p><b>Title</b> - The title of the content block</p> <p><b>ID</b> - The XML content block ID number</p> <p><b>Date Created</b> - The date the content block was created</p> <p><b>Date Modified</b> - The date the content block was last modified</p> <p><b>LastEditorLName</b> - The last editor's last name</p> <p><b>LastEditorFname</b> - The last editor's first name</p> <p><b>UserId</b> - The ID of the user who last edited the content</p>	String	
OrderDirection	<p>The direction the search results will be sorted in. The default is <b>Ascending</b></p> <p><b>Ascending</b> - Results from A to Z</p> <p><b>Descending</b> - Results from Z to A</p>	String	
PostResultsAt	<p>Enter the Web form on which the search results will be posted.</p> <p>If this property is left blank, the results are posted to the Web form from which the search was initiated.</p> <hr/> <p><b>Important!:</b> A search control must exist on the posting page.</p> <hr/>	String	
Recursive	Whether to search subfolders of the identified root folder. The root folder is identified in the <code>FolderID</code> property.	Boolean	See Also: <a href="#">"FolderID" on page 327</a>
RemoveTeaserHtml	<p>An optional search result formatting parameter that removes all the HTML from each content item's teaser/summary before sending it to the browser.</p> <p><b>True</b> = Remove HTML</p> <p><b>False</b> = Do not remove HTML</p>	Boolean	

Property	Description	Data Type	For more information, see
ResultTagId	Specify the div tag control ID to display the search results. This property is only active when the EnableAjax property is set to true.	String	<a href="#">"ResultTagId" on page 332</a> <a href="#">"EnableAjax" on page 323</a>
ShowDate	If <b>True</b> , displays the content block's last modified date. This is useful for determining the most recent content block.	Boolean	<a href="#">"ShowDate Examples" on page 336</a>
ShowExtendedSearch	<p>Allows a user to search via Searchable Property metadata.</p> <p><b>True</b> = Show extended search</p> <p><b>False</b> = Hide extended search</p> <p>Search type metadata only appears on the search screen if the following are true</p> <ul style="list-style-type: none"> <li>• this property is set to <b>True</b></li> <li>• the ShowSearchOptions property is set to <b>True</b></li> <li>• at least one content item in the starting folder (and its subfolders if <code>recursive</code> is set to true) has searchable metadata applied to it</li> </ul> <hr/> <p><a href="#">Note: The folderID property determines the search's starting folder.</a></p> <hr/>	Boolean	<a href="#">"ShowExtendedSearch" on page 336</a> For more information on Metadata and Searchable Property, see the Administrator Manual section "Metadata".
ShowSearchBoxAlways	<p>If set to <b>false</b>, the search box does not appear on PostBack. The default is True.</p> <p><b>True</b> = Display search box on PostBack</p> <p><b>False</b> = Do not display search box on PostBack</p>	Boolean	



Property	Description	Data Type	For more information, see
ShowSearchOptions	<p>If set to true, the following choices appear along with the search box. This type of search is also known as a Standard search. The default setting is True.</p> <ul style="list-style-type: none"> <li>• All the words</li> <li>• Any of the words</li> <li>• Exact phrase</li> <li>• Match Partial words</li> </ul> <p><b>True</b> = Show search options  <b>False</b> = Hide search options</p>	Boolean	<a href="#">"Basic and Standard Searches" on page 340</a>
Stylesheet	<p>Specify the location and style sheet for the search results. For example:  <code>csslib\mytest.css</code>.            Leave blank to use the default.  <code>global.css</code> is the default style sheet. It is located in <code>\webroot\Workarea\csslib</code>.</p>	String	<a href="#">"Using a Style Sheet" on page 349</a>
SuppressWrapperTags	<p>Suppresses the output of the span/div tags around the control. The default is <b>False</b>.  <b>True</b> - Suppress wrap tags.  <b>False</b> - Allow wrap tags.</p>	Boolean	
Text Box Size	<p>The size of the display text box for user input. The width is based on the number of characters.</p>	Integer	<a href="#">"Text Box Examples" on page 333</a>
WrapTag	<p>Allows a developer to specify a server control's tag.            The default is <b>Span</b>.  <b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element.  <b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.  <b>Custom</b> - Allows you to use a custom tag.</p>	String	

## Ajax Search

Ajax gives your site visitors an application-like search experience. For example, the search displays results with reloading the page. In addition, results can appear in a separate area of the Web form.

The Ajax search also lets you implement a paging feature. When `EnableAjax` is set to `true`, the `MaxResults` property determines the maximum number of results per page. See Also: ["MaxResults" on page 328](#).

**SEARCH**

Search Text:

Total results: 5  
Viewing page 1 of 2  
[First] [Previous] [Next] [Last]

Business Practices	ektron Medical's commitment to operating with integrity requires more than just adequate financial m...
Contact Form	
Ektron Rated Positive	Ektron Inc., an innovator in Web content management software, today announced ...

### ResultTagId

By using the `ResultTagId` property, you can designate where you want search results to appear. This allows you to place the search criteria in one area of a Web form and the results in another. The `ResultTagId` property only works with the Ajax Search.

For example, you have the following Tag.

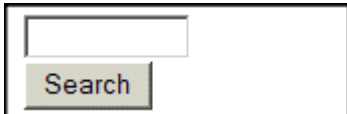

```

```

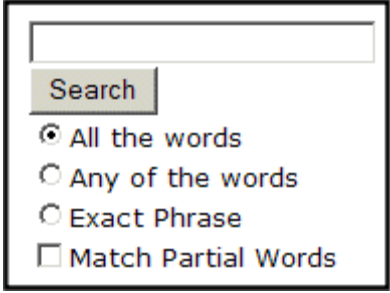
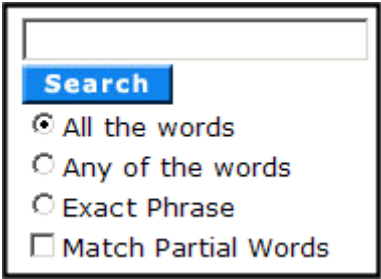
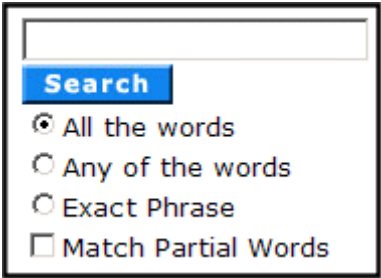
You would add **results** to the property, as shown below.

Recursive	True
RemoveTeaserHtml	True
ResultTagId	results
ShowDate	False
ShowExtendedSearch	True

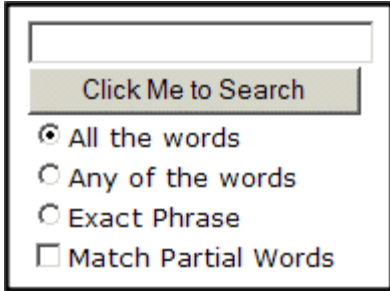
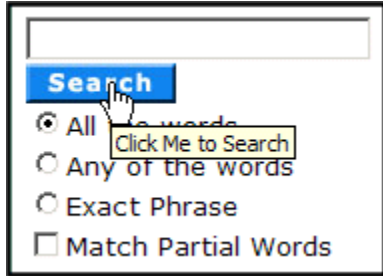
## Text Box Examples

Value	Web Page View
10	
50	

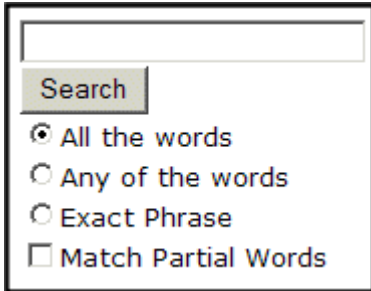
## Button Image Source Examples

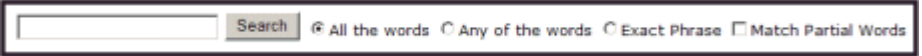
Value	Web Page View
(blank)	
search.jpg	
http://www.example.com/images/search.jpg	

## Search Button Text Examples


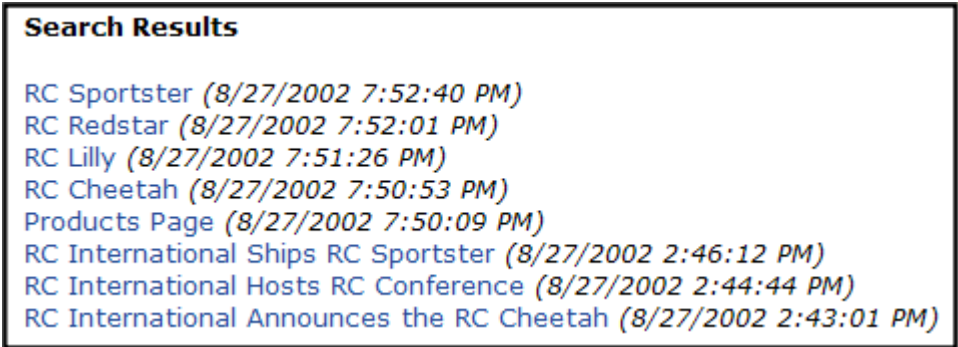
Value	Web Page View
"Click Me to Search"	 <p>The image shows a search interface with a text input field at the top. Below the field is a button labeled "Click Me to Search". Underneath the button are four radio button options: "All the words" (selected), "Any of the words", "Exact Phrase", and "Match Partial Words" (which is a checkbox).</p>
"Click Me to Search"	 <p>The image shows a search interface with a text input field at the top. Below the field is a blue button labeled "Search". A mouse cursor is hovering over the button, and a tooltip box appears with the text "Click Me to Search". Underneath the button are four radio button options: "All the words" (selected), "Any of the words", "Exact Phrase", and "Match Partial Words" (which is a checkbox).</p>

## Display Examples

Value	Web Page View
Vertical	 <p>The image shows a search interface with a text input field at the top. Below the field is a button labeled "Search". Underneath the button are four radio button options: "All the words" (selected), "Any of the words", "Exact Phrase", and "Match Partial Words" (which is a checkbox).</p>

Value	Web Page View
Horizontal	

## ShowDate Examples

Value	Web Page View
False	
True	

## ShowExtendedSearch

ShowExtendedSearch lets you decide if the user can search the Searchable Property contained in the metadata of a content block. If you set the ShowExtendedSearch property to **true**, and a site visitor visits the search page, the extended search parameters appear below the search form.

Search

Text:





☒ All the words

☐ Any of the words

☐ Exact Phrase

☐ Match Partial Words [Advanced](#)

Audio Title:

Image Date: between [None]   and [None]  

Part Number: between  and

The Searchable Properties for a folder must be turned on for them to appear in a search. For more information on using Searchable Properties, see the Administrator Manual section "Metadata."

The examples below show Searchable Properties being activated in the folder properties and appearing on a search page.

☐ Publish Office documents as PDF

**Meta Data/Custom-Fields available for folder 'Content':**

Assigned	Required	Name
<input checked="" type="checkbox"/>	<input type="checkbox"/>	title
<input checked="" type="checkbox"/>	<input type="checkbox"/>	keywords
<input checked="" type="checkbox"/>	<input type="checkbox"/>	description
<input checked="" type="checkbox"/>	<input type="checkbox"/>	page
<input checked="" type="checkbox"/>	<input type="checkbox"/>	robots
<input checked="" type="checkbox"/>	<input type="checkbox"/>	breadcrumb
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Audio Title
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Image Date
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Part Number

**Notification Settings:**

Address http://192.168.0.82/CMS400Example/webform5.aspx

Search

Text:

☒ All the words  
☐ Any of the words  
☐ Exact Phrase  
☐ Match Partial Words [Advanced](#)

Audio Title:

Image Date: between [None] and [None]

Part Number: between  and



## Using the Advanced Search Properties

There are three ways to use the advanced search properties. One way is to create a basic or standard search, in which the user must search by properties you choose. Another way is let users decide which properties they want use for the search. The last way is a mix of both. See Also: ["Basic and Standard Searches" on page 340](#), ["Advanced Search" on page 341](#) and ["Using Standard and Advanced Searches Together" on page 342](#)

The following is a list of advanced properties. These properties appear in the list of properties for the server control. For descriptions of these properties, see ["Search Server Control Properties" on page 319](#).

- EnableArchived
- EnableArchivedBtn
- EnableContent
- EnableContentBtn
- EnableDMSAssets
- EnableDMSAssetsBtn
- EnableForms
- EnableFormsBtn
- EnableLibFiles
- EnableLibFilesBtn
- EnableLibHyperlinks
- EnableLibHyperlinksBtn
- EnableLibImages
- EnableLibImagesBtn

---

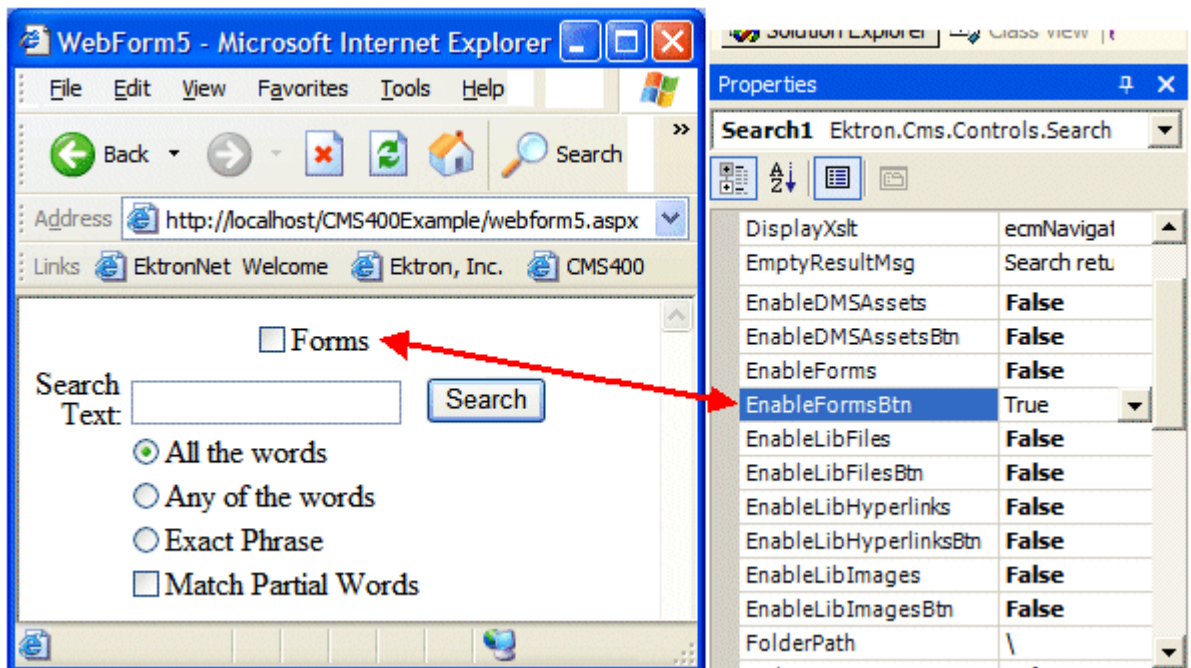
**WARNING!** If you set all above properties to **false**, the search returns no results. This happens because there are no parameters to search against.

---

These properties can be used individually or in combination. For example, you could have a search that only searches Forms, or a search that searches Forms and DMS Assets.

Each advanced property has an **enable** check box property. These properties have **Btn** at the end of their name. They create a check box that appears when the Advanced link on a search is clicked. The check boxes only appear when the properties are set to true. Using the Advanced link and check boxes to search is known as an Advanced Search. See Also: ["Advanced Search" on page 341](#)

The following example shows the Forms check box enabled on the site's search page and the EnableFormsBtn property set to True in Visual Studio.



You can force a user to search using only properties you define, preventing him from changing any properties. For example, to create a forms-only search, set all properties with **Enable** in the name to **false** except for EnableForms. This is known as a standard search. See Also: ["Basic and Standard Searches" on page 340](#)

**NOTE** When setting up a server control for a standard search, you must choose the actual property to search with, not the property that enables the check box. For example, to search forms, use the EnableForms property, not the EnableFormsBtn property.

## Basic and Standard Searches

A basic search contains only three items when displayed on a page:

- text to describe the text box
- a search text box
- a submit or "search" button

A basic search takes the least amount of real estate on your page and provides the fewest number of options to a site visitor. Below is an example of a basic search. This is handy when you want to have a search perform specified functions with little user control. To make your search control appear as a basic search, set the ShowSearchOptions property to **false**.

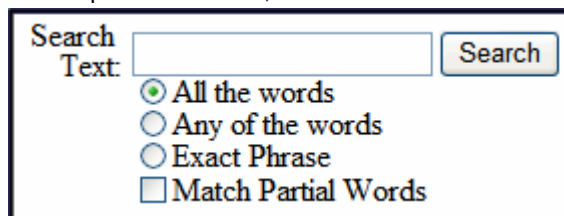
A standard search control consisting of a text input field with the label "Search Text" and a "Search" button.

A standard search allows a site user a few more options when searching your site. These options, listed below, allow a site user to decide how the text should match the information they are searching for.

- All the words
- Any of the words
- Exact phrase
- Match Partial words

With basic and standard searches, you decide which properties are used in the search. At least one advanced search property must be set to **true** in the server control. Otherwise, the search returns no matches.

When added to a template and visited, a standard search looks like this.

A standard search control with a text input field labeled "Search Text" and a "Search" button. Below the input field are four radio button options: "All the words" (selected), "Any of the words", "Exact Phrase", and "Match Partial Words".

Users enter text, change the way the text matches the items being searched, and click the search button to search for the types of content you allow. For examples of setting up a standard search, see ["Using Standard and Advanced Searches Together" on page 342](#)

## Advanced Search

In an advanced search, you decide which content types are available. From those choices, site visitors decide which content types they want and perform a search.

To let a site visitor choose which content types to search, the Advanced Hyperlink must be turned on. Otherwise, users can only search properties you have enabled. See Also: ["EnableAdvancedLink" on page 323](#)



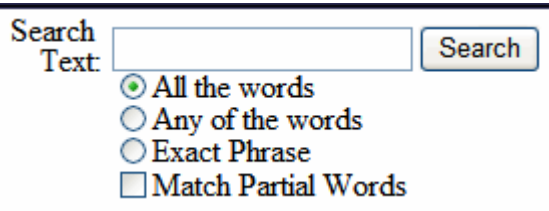
When the Advanced hyperlink is enabled, this is how the search appears.

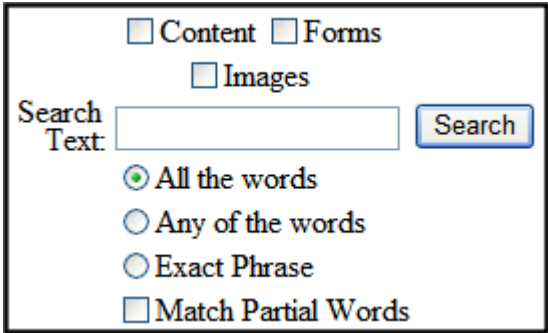
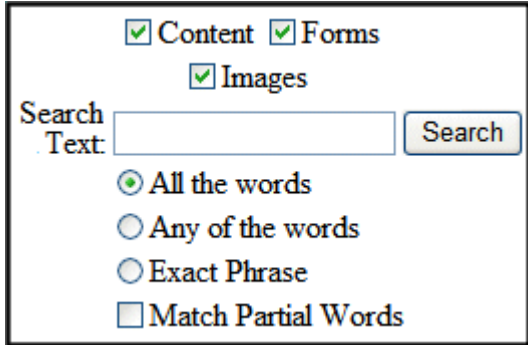
When the Advanced hyperlink is clicked, this is how the search appears. This assumes all choices are enabled and automatically checked. For examples of setting up an advanced search, see ["Using Standard and Advanced Searches Together" on page 342](#)

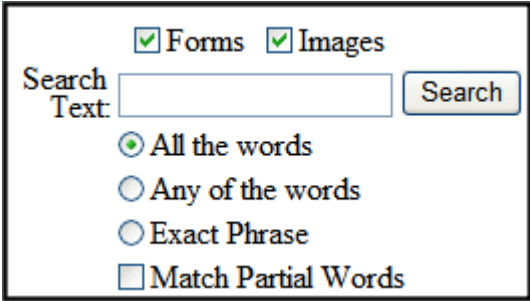
## Using Standard and Advanced Searches Together

You can use a mix of the standard search and advanced search to force people to search by certain properties while giving them a choice on others. For example, you want users to always search Content, yet give them an option to search forms, files, and images.

The following examples illustrate how to configure the Search server control using the advanced properties. If a property is not mentioned in the examples below, use its default setting.

Description	How the Search Appears to Site Visitors	Property Settings
<p>Search content only. Do not let the site user change advanced settings. No standard options appear.</p> <p>This is a <b>Basic search</b>. See Also: <a href="#">"Basic and Standard Searches"</a> on page 340</p>		<p>EnableAdvancedLink = False</p> <p>EnableContent = True</p> <p>Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>
<p>Search content, forms, and library images only. Do not let the user change advanced settings. No standard options appear.</p> <p>This is a <b>Basic search</b>. See Also: <a href="#">"Basic and Standard Searches"</a> on page 340</p>		<p>EnableAdvancedLink = False</p> <p>EnableContent = True</p> <p>EnableForms = True</p> <p>EnableLibImages = True</p> <p>Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>
<p>Search content only. Do not let the site user change advanced settings. Standard options appear.</p> <p>This is a <b>Standard search</b>. See Also: <a href="#">"Basic and Standard Searches"</a> on page 340</p>		<p>EnableAdvancedLink = False</p> <p>EnableContent = True</p> <p>ShowSearchOptions = True</p> <p>Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>

Description	How the Search Appears to Site Visitors	Property Settings
<p>Let the user choose whether to search content, forms, library images, or all of them together. The user can also mix and match properties. Standard options appear.</p> <p>For example, search content and library images, but not forms.</p> <p>This is an <b>Advanced search</b>. See <i>Also:</i> <a href="#">"Advanced Search" on page 341</a></p> <hr/> <p><b>Warning:</b> If a user does not check at least one box, the search returns no results.</p> <hr/>	 <p>The screenshot shows a search form with three unselected checkboxes at the top: <input type="checkbox"/> Content, <input type="checkbox"/> Forms, and <input type="checkbox"/> Images. Below these is a text input field labeled "Search Text:" followed by a "Search" button. Underneath the input field are four radio button options: <input checked="" type="radio"/> All the words, <input type="radio"/> Any of the words, <input type="radio"/> Exact Phrase, and <input type="checkbox"/> Match Partial Words.</p>	<p>EnableAdvancedLink = True  EnableContentBtn = True  EnableFormsBtn = True  EnableLibImagesBtn = True  ShowSearchOptions = True  Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>
<p>The check boxes are checked by default. The user can change the check boxes. Standard options appear.</p> <p>This is an <b>Advanced search</b>. See <i>Also:</i> <a href="#">"Advanced Search" on page 341</a></p> <hr/> <p><b>Warning:</b> If a user unchecks all of the boxes, the search returns no results.</p> <hr/>	 <p>The screenshot shows a search form with three selected checkboxes at the top: <input checked="" type="checkbox"/> Content, <input checked="" type="checkbox"/> Forms, and <input checked="" type="checkbox"/> Images. Below these is a text input field labeled "Search Text:" followed by a "Search" button. Underneath the input field are four radio button options: <input checked="" type="radio"/> All the words, <input type="radio"/> Any of the words, <input type="radio"/> Exact Phrase, and <input type="checkbox"/> Match Partial Words.</p>	<p>EnableAdvancedLink = True  EnableContent = True  EnableContentBtn = True  EnableForms = True  EnableFormsBtn = True  EnableLibImages = True  EnableLibImagesBtn = True  ShowSearchOptions = True  Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>

Description	How the Search Appears to Site Visitors	Property Settings
<p>The user is forced to search Content but can choose to search Forms or Images. The Forms and Images check boxes are checked by default. Standard options appear.</p> <p>This is a mix of Standard and Advanced searches. See Also: <a href="#">"Using Standard and Advanced Searches Together"</a> on page 342</p> <hr/> <p><b>Warning:</b> If a user unchecks all of the boxes, the search still search content and will return any matching results.</p> <hr/>		<p>EnableAdvancedLink = True  EnableContent = True  EnableForms = True  EnableFormsBtn = True  EnableLibImages = True  EnableLibImagesBtn = True  ShowSearchOptions = True  Set all other properties with the word <b>Enable</b> to <b>False</b>.</p>

## Fieldset Properties

**NOTE** The properties described below can be seen by a site visitor only when the Advanced link is enabled and clicked.

The fieldset properties let you refine the look and feel of your site's search by separating items used in the Advanced search into groups. Each group can have its own legend. The legends makes it easier for site visitors to understand their choices and how those choices affect the search.

The fieldset properties are:

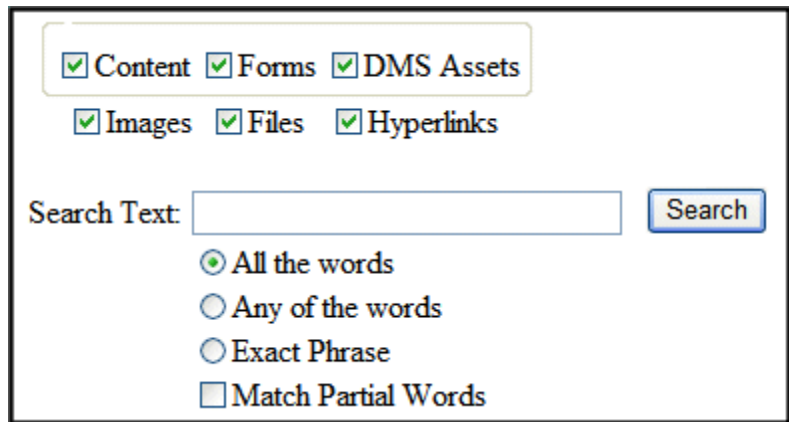
- ContentFieldsetLegend
- EnableContentFieldset
- FolderFieldsetLegend
- EnableFolderFieldset
- LibraryFieldsetLegend

- EnableLibraryFieldset

These properties can be broken down into two types:

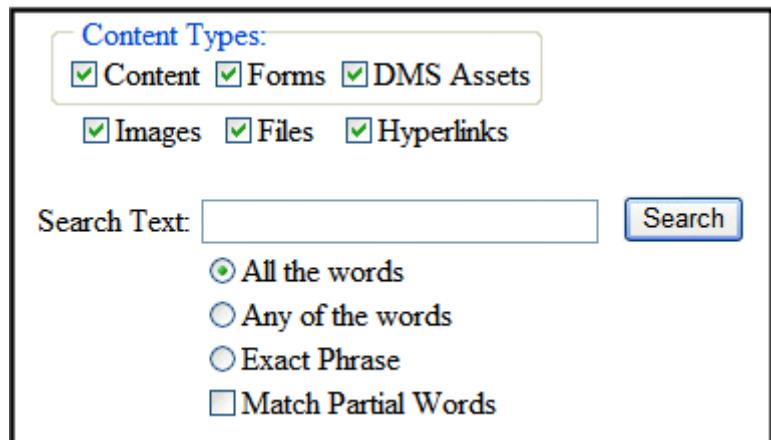
- enabling fieldsets
- enabling fieldset legends

When you enable a fieldset, a border is created around a group of items. For example, if the `EnableContentFieldset` property is set to **true**, a border surrounds the different types of content.



A screenshot of a search control interface. At the top, there is a rounded rectangle border containing six checkboxes, each with a green checkmark: ☒ Content, ☒ Forms, ☒ DMS Assets, ☒ Images, ☒ Files, and ☒ Hyperlinks. Below this border, there is a "Search Text:" label followed by a text input field and a "Search" button. Under the input field, there are four radio button options: ☒ All the words, ☐ Any of the words, ☐ Exact Phrase, and ☐ Match Partial Words.

You can add a legend to the fieldset to describe its content. For example, by adding **Content Types:** to the `ContentFieldsetLegend` property, you create a legend for the fieldset around the different types of content.



A screenshot of a search control interface, similar to the one above but with an added legend. The legend, "Content Types:", is written in blue text and is positioned above the rounded rectangle border that contains the six checked checkboxes: ☒ Content, ☒ Forms, ☒ DMS Assets, ☒ Images, ☒ Files, and ☒ Hyperlinks. Below the border, there is a "Search Text:" label followed by a text input field and a "Search" button. Under the input field, there are four radio button options: ☒ All the words, ☐ Any of the words, ☐ Exact Phrase, and ☐ Match Partial Words.



## Retrieving the XML Structure of a Search

Retrieving the XML structure of XML content provides greater control over developing XSLs. The following is an example of how to retrieve the XML structure:

1. Open a new Web form.
2. Drag and drop a Search server control onto it.
3. Set the FolderID property.
4. Drag and drop a Textbox on the Web form.
5. Set the TextMode property to MultiLine.

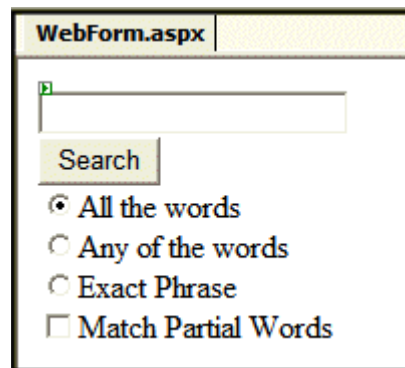
**NOTE** It is also recommended that you set the width of the text box to at least 400px.

6. On the code behind page, add the following line.  
`Textbox1.Text = Search1.XmlDoc.InnerXml`
7. Build the project.
8. View the Web form in a browser.
9. Search for something.
10. When the search returns any items, the XML structure of those items appear in the textbox.

## Displaying Search Results on a Separate Page

The following example shows a search dialog box on a Web form, and the results displayed on separate Web form. This is done when you the search dialog box appears in a navigation pane, and the results appear on a separate Web form.

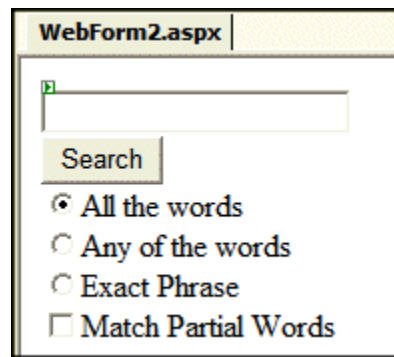
1. Drag and drop a Search server control on a Web form.



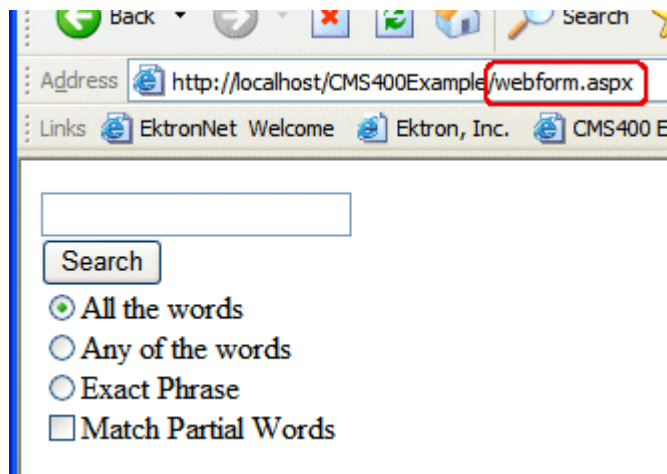
2. Set the following Ektron properties for the Search server control.
  - FolderID** = choose the ID of the folder to search. 0 (zero) search from the root.
  - EnableAjax** = False
  - DisplayXslt** = None - Databind Only
  - PostResultAt** = The Web form that displays the search results. For example, `Webform2.aspx`

**NOTE** `webform2.aspx` represents the name of the second Web form.

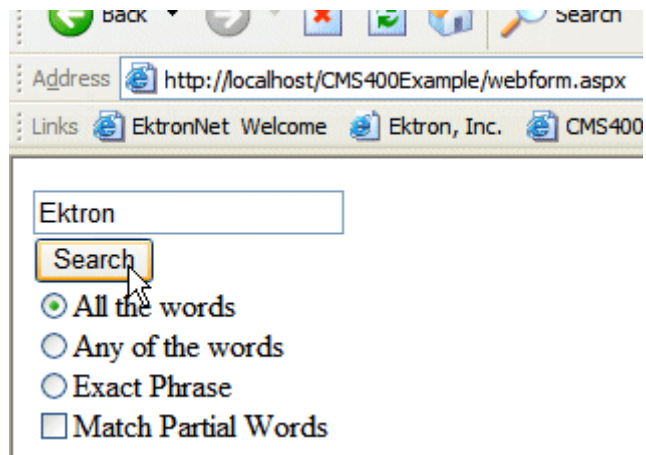
3. Set any additional properties as needed. You do not need to set any more properties than ones listed in the previous step for this example to work. This step is optional.
4. Add a second Web form to the project.
5. Drag and drop a Search server control on it.



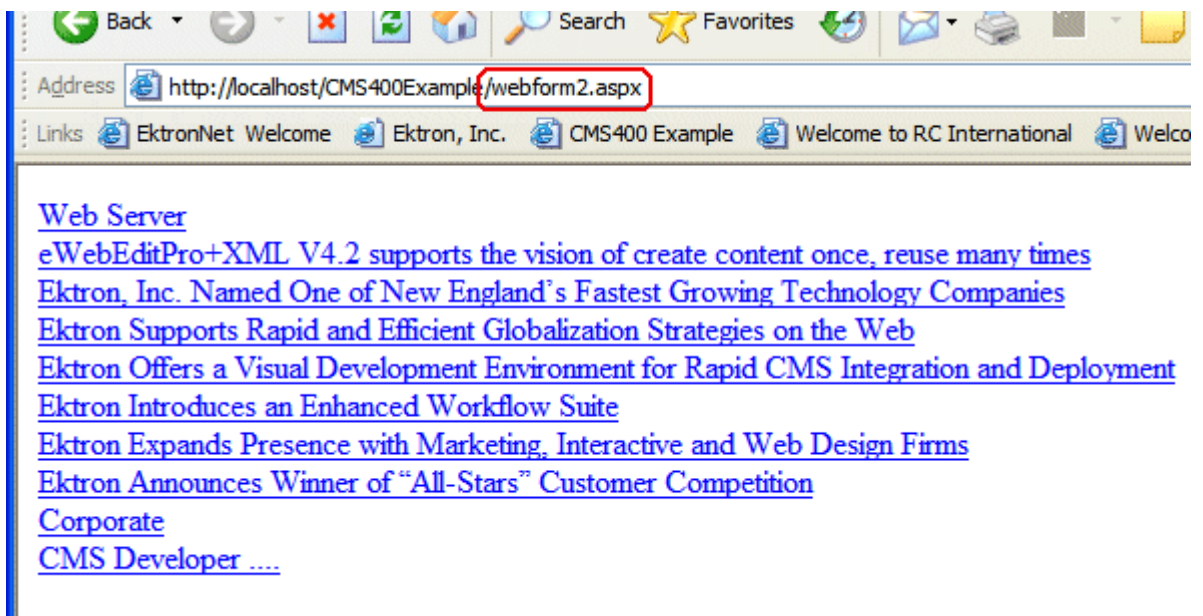
6. Set the Ektron property **ShowSearchBoxAlways** to **false**.
7. Build the solution.
8. In your browser, go to the first Web form.



9. Use the search dialog.



10. The search results display on the second Web form.



## Using a Style Sheet

You can create your own style sheet, or modify the existing default one, `global.css`. It is located in `\webroot\siteroot\Workarea\csslib`.

Style Sheets let you choose different text colors, text alignment, text fonts, and text size to create a customized feel for your search.

**Best Practice**

Ektron recommends copying and renaming the default style sheet. This insures you always have a clean file to start with. This also gives you something to reference if you are not getting the results you expect.

The style sheet must be located in the site root.

You can only define one Search style sheet for each Web page. If you have more than one Search on a single Web page, they share the style sheet.

For more information on style sheets, visit <http://www.w3.org/Style/CSS/>.

## Formatting the Search Server Control

Additional information on formatting the Search server control can be found in the following Knowledge Base article.

<http://www.ektron.com/developers/cms400kb.cfm?id=2322>

# IndexSearch Server Control

Use the IndexSearch server control to display a Smart Form's search screen on any Web form. This lets a user search your site based on XML parameters.

See *Also*: The **Ektron CMS400.NET** Administrator Manual section "XML Indexing".

Below is a sample screen that can be generated via the IndexSearch server control.

The image shows a web form with the following elements:

- Genre:** A dropdown menu with 'novel' selected.
- Price:** A dropdown menu with 'Between' selected, followed by a text input containing '1' and an empty text input.
- Last Name:** A dropdown menu with 'Jones' selected.
- Paperback:** An unchecked checkbox.
- Publishdate:** A dropdown menu with 'Select' selected, followed by a text input and a small calendar icon.
- Title:** A dropdown menu with 'Contains' selected, followed by a text input containing 'cms'.
- Submit Search:** A button at the bottom right of the form.

You can also predefine a search and have the results appear on a Web form template by using the SearchParmXML property.

This subsection contains the following topics:

- ["IndexSearch Server Control vs. Search Server Control" on page 351](#)
- ["IndexSearch Server Control Properties" on page 351](#)
- ["SearchParmXML Property" on page 358](#)
- ["Using the IndexSearch Server Control Programmatically Example" on page 361](#)
- ["Programmatically Predefined General Search Results Replaced By Specific Search Results" on page 364](#)
- ["Turn Off Caching In Netscape Navigator and Mozilla Firefox" on page 367](#)

## IndexSearch Server Control vs. Search Server Control

See, ["Search Server Control vs. IndexSearch Server Control" on page 319.](#)

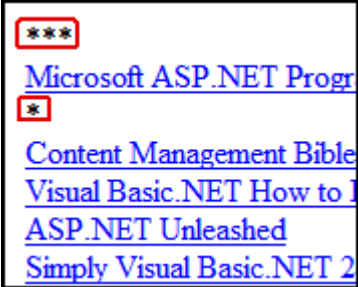
## IndexSearch Server Control Properties

The IndexSearch server control properties are described in this table.

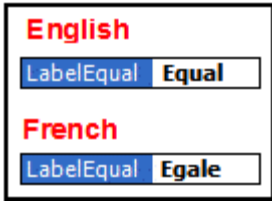
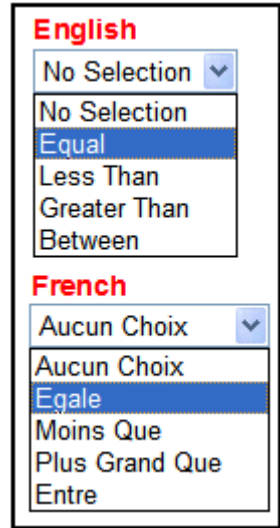
**NOTE** The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Description	Data Type	Values
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to the folder needed for the <b>FolderID</b> property. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>	String	
ButtonImgSrc	If you want to display an image on the submit button, enter the server path to it. See Also: <a href="#">"Button Image Source Examples" on page 334.</a>	String	
ButtonText	The text that appears on the button.	String	The default is: Search
ContentParameter	Checks the QueryString for the content block ID value and replaces the search with content when ID is specified. Leave blank to always display the search.	String	None - Use Default ID - The server control uses the content block passed as a URL parameter.

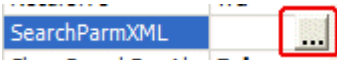
Property	Description	Data Type	Values
DisplayXslt	Determines how the information on the page is displayed	String	<p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder</p> <p>See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary</p> <p>See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page</p>
EmptyResultMsg	The message that displays if the search returns no hits.	String	The default value is: Your search did not match any documents.
EncodeContentHtml	When set to True, Content HTML is encoded when it is returned.	Boolean	<p><b>True</b> = Content HTML is encoded</p> <p><b>False</b> = Content HTML is not encoded</p>
FolderId	<p>The numeric id of the folder that you want to search. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a></p> <p>The Recursive property determines whether the IndexSearch server control will use this folder's child folders in the search.</p>	Integer	<p>Any folder's numeric ID</p> <p><b>0</b> (zero) represents the root folder.</p>

Property	Description	Data Type	Values
Hide	Used to hide output of the index search in design time and run time. For example, you want to pass the results to code behind for further manipulation.	Boolean	<b>True</b> = Hide results <b>False</b> = show results
HitsMarker	<p>This symbol shows how many search criteria each hit matched.</p> <p>The following example shows the first hit met three of the criteria. The second group met one criterion.</p>  <p>This property is used when the weighted property is set to true. See Also: "Weighted" on page 358.</p>	String	<p>* # +</p> <p><b>(Custom)</b> Custom lets you choose the marker to use. The custom marker can be any text, numeric, or symbol character except &amp; and &lt;.</p> <p>For more Information, see The Ektron CMS400.NET Administrator Manual section "XML Indexing" &gt; "Displaying Search Results"</p>



Property	Description	Data Type	Values
Labelafter Labelbefore LabelBetween LabelContains LabeldateBetween LabelEqual LabelExactPhrase LabelGreaterThan Labelinputdate Labelinputnumber LabelLessThan Labelon LabelSelect	<p>All of these properties let you change their label names. This is useful for Multi-language issues. For example, if you have a French site, you change <b>Equal</b> to <b>Égale</b>.</p>  <p>Note: These label properties will not change based on the <b>Language</b> property's setting. For example, setting the Language property to 1036 will not change these labels to French.</p>	String	<p>Examples of what the user will see on the site in French and English.</p> 
Language	Set a language for viewing index search results. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer	

Property	Description	Data Type	Values
LinkTarget	Defines the way a link acts when a link is clicked.	String	<p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>
MaxResults	The Maximum number of content blocks returned (0=unlimited).	Integer	

Property	Description	Data Type	Values
OrderBy	The order of the list to be returned. For example, if you want to sort search results by last modified date, choose <b>Date Modified</b> .	String	<ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>ID</b> - The XML content block ID number</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>AuthorLName</b> - The last author's last name</li> <li>• <b>AuthorFName</b> - The last author's first name</li> <li>• <b>StartDate</b> - The GoLive Date</li> </ul>
OrderByDirection	The direction the results will be sorted in. This property works with the <b>OrderBy</b> property only.	String	Ascending Descending
Recursive	Determines whether the IndexSearch server control will use this folder's child folders in the search.	Boolean	True False
SearchParmXML	<p>Opens a separate window that lets you predefine a search. For example, you want a predefined list of books to appear on a web page.</p> <p>To access the window click the gray button.</p> 		For more information on SearchParmXML, see <a href="#">"SearchParmXML Property" on page 358</a>
ShowSearchBoxAlways	If set to <b>false</b> , the search box does not appear onPostBack	Boolean	True False

Property	Description	Data Type	Values
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is <b>False</b> .	Boolean	<b>True</b> - Suppress wrap tags. <b>False</b> - Allow wrap tags.
Weighted	<p>If you set this value to <b>true</b>, content blocks are ranked by the number of criteria matches.</p> <p>For example, if a content block matches all search criteria, it appears at the top of the results page. A hit marker appears for each match.</p> <p>If a second content block only matches one criterion, it also appears on the results page underneath the first content block. One asterisk appears above the page's title to indicate the single match.</p>	Boolean	<p><b>True</b> - display every content block that matches <i>at least one</i> search criteria</p> <p><b>False</b> - show only content blocks that match <i>all</i> search criteria</p>
WrapTag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p>	String	<p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p> <p><b>Custom</b> - Allows you to use a custom tag.</p>
xmlConfigID	ID of the Smart Form whose indexed fields will appear on the search screen.	Integer	

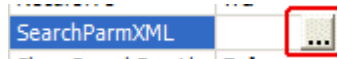
## SearchParmXML Property

The SearchParmXML property is used to create a predefined XML search, whose results appear upon page load. Clicking on the gray box in this property opens a window, where you define the search. The parameters in this window are defined by the xmlConfigID you choose. Using this property launches the CMS Explorer login, if you are not already logged in. For more information on the CMS Explorer

login, see ["Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15](#)

The following example uses the SearchParmXML property to create a list of books, based on publishers with "press" in their name.

1. To activate the window, click the gray box in the SearchParmXML property.



2. The Build Search Parameter screen appears.

**Build Search Parameter**

**Search Parameter**

☐ /root/publisher string

Select One

**Search Parameter**

☐ /root/pdate date

Select One

**Search Parameter**

☐ /root/price decimal

Select One

**Search Parameter**

☐ /root/authors/author/lastname string

Select One

- Click the check boxes for the parameters you want to search with.

The screenshot shows the 'Build Search Parameter' dialog box with the following parameters:

- ☒ **/root/publisher** (string) - Select One
- ☐ **/root/pdate** (date) - Select One
- ☐ **/root/price** (decimal) - Select One
- ☐ **/root/authors/author/lastname** (string) - Select One

- Define the parameter.

The screenshot shows the 'Build Search Parameter' dialog box with the following parameter:

- ☒ **/root/publisher** (string) - Contains

The text input field contains 'Press'.

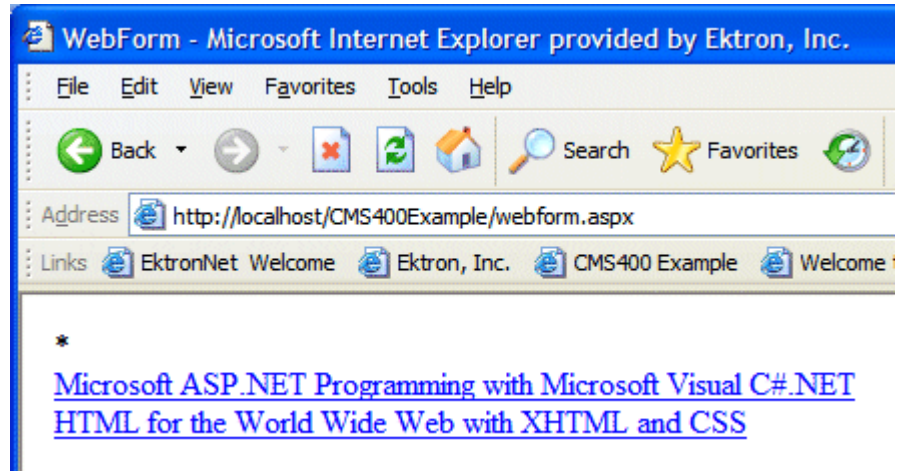
The drop down box in this example contains the following variables:

**Select One** - Prompts you to choose a variable.

**Exact Phrase** - You enter a phrase and search for that exact phrase, contained within the XML content.

**Contains** - You enter a word or phrase and search for any XML content, that has that word or phrase contained within it.

- Click OK.
- Build the Web form.
- Browse to the Web page in your browser.



8. The page appears with a list of XML content blocks that match the parameters you set.

## Using the IndexSearch Server Control Programmatically Example

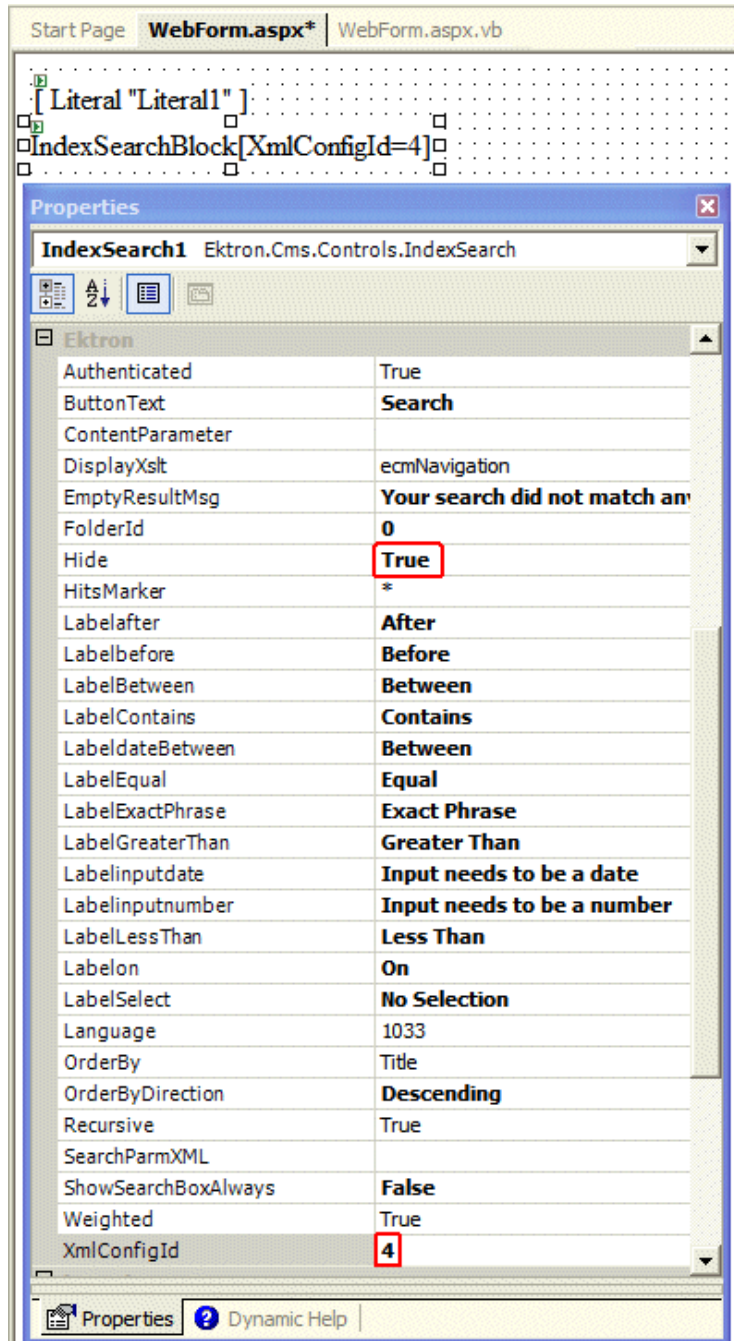
**NOTE** When creating an IndexSearch server control form user interface using code behind only, you need the IndexSearch server control to be connected with the page events. Otherwise you will not be able to submit your search parameters.

You can use the IndexSearch server control, in code behind, to manipulate the way users view the search results. The following example, using code behind, shows how to return values associated with an XPath. This example shows a list of book subject titles.

1. Open a Web form.
2. Drag and drop an IndexSearch server control on your Web form.
3. Set the properties in the properties window.

**NOTE** In this example, the hide setting must be set to **true**. Remember to set the XmlConfigID. This example uses ID number **4**.

4. Drag and drop a Literal on the Web form.



5. Add the following code to the code behind.

```
IndexSearch1.Parm.XPath = "/root/subject"
Dim arStr As String()
Dim Str As String
```



```

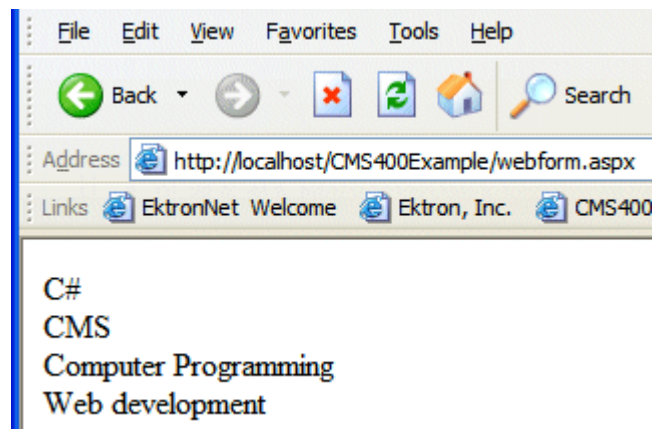
Dim strRet As String
arStr = IndexSearch1.GetXPathValues("/root/subject")
For Each Str In arStr
 strRet += Str & "
"
Next
Literal1.Text = strRet

```

Here is a breakdown of the code.

Code snippet	Description
IndexSearch1.Parm.XPath = "/root/subject"	Defines the parameter XPath location:
Dim arStr As String() Dim Str As String Dim strRet As String	Creates variables
arStr = IndexSearch1.GetXPathValues("/root/subject")	Sets the variable arStr as an array of the XPathValues
For Each Str In arStr strRet += Str & " " Next	The for next loop iterates through the array of XPathValues
Literal1.Text = strRet	Displays values on your Web form

6. Build the project.
7. Browse to your web form.
8. The list of book subject titles appears.



## Programmatically Predefined General Search Results Replaced By Specific Search Results

This Code behind example shows a predefined search appearing on a Web page, then being replaced by more specific search results. This example creates a hyperlink list of book publishers. When you click a specific publisher, it replaces the publisher list with a book list from that publisher.

Follow these steps.

1. Add an IndexSearch server control to a Web form.
2. Set the following properties:
  - DisplayXslt = ecmTeaser
  - Hide = True
  - HitsMarker = (blank)
  - ShowSearchBoxAlways = False
  - Weighted = True
  - XmlConfigId = 4
3. Add two Literals.
4. Add the following code to the Page Load Event in code behind.

```
Dim bRet As Boolean
If Request.QueryString("publisher") <> "" Then
 IndexSearch1.Hide = False
 IndexSearch1.Parm.XPath = "/root/publisher"
 IndexSearch1.Parm.DataType =
Ektron.Cms.Common.EkEnumeration.XMLDataType.String
 IndexSearch1.Parm.SearchType =
Ektron.Cms.Common.EkEnumeration.XMLSearchRangeType.ExactPhrase
 IndexSearch1.Parm.Value1 = Request.QueryString("publisher")
 bRet = IndexSearch1.AddParm()
 IndexSearch1.Search()

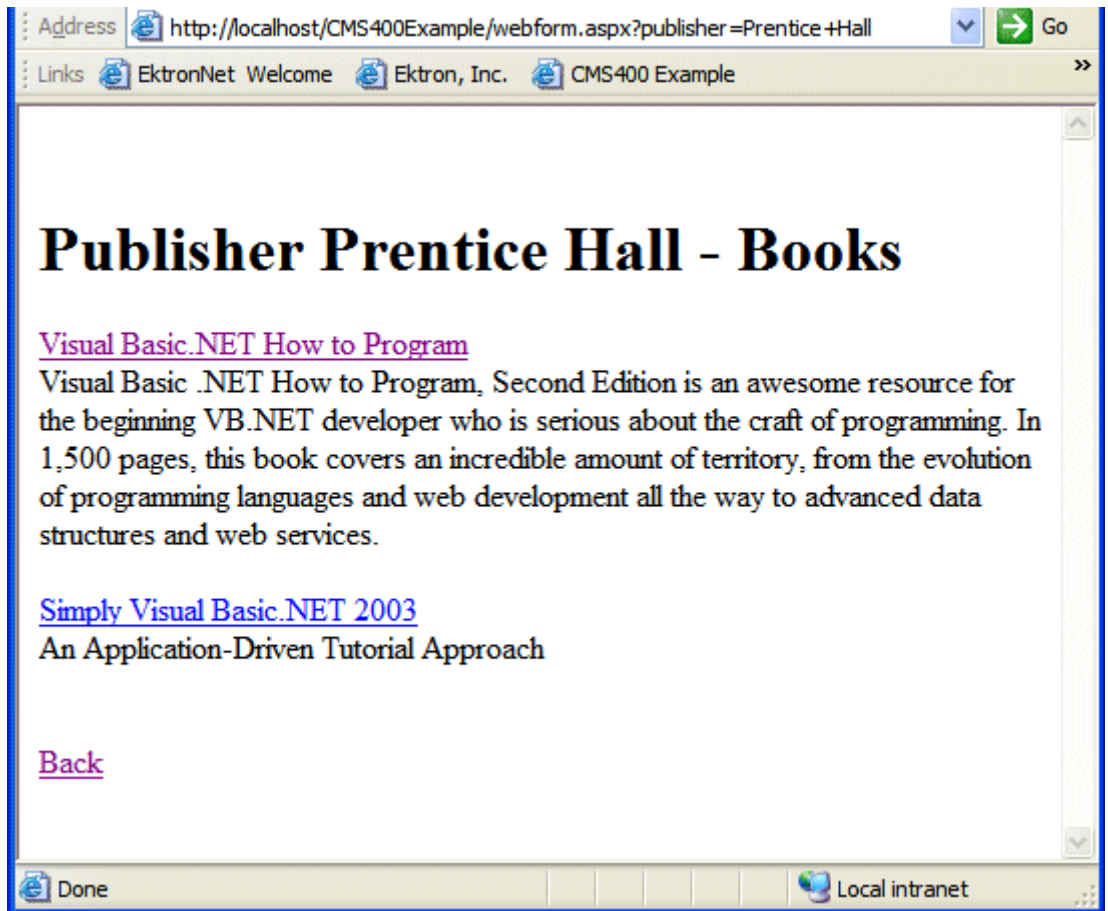
 Literal1.Text = "<h1>Publisher " & Request.QueryString("publisher") & " -
Books</h1>"
 Literal2.Text = "Back"
Else
 Literal1.Text = "<h1>Publishers</h1>"
 Dim arStr As String()
 Dim Str As String
 Dim strRet As String
 arStr = IndexSearch1.GetXPathValues("/root/publisher")
 For Each Str In arStr
 strRet += "<a href=\""webform.aspx?publisher=\"" & Server.UrlEncode(Str) &
"">" & Str & "
"
 Next
 Literal1.Text += strRet
End If
```

5. Save your Web form.
6. Build your Web form.
7. Browse to the Web form in your browser.
8. The list of publishers appears.



9. Click on a publisher.

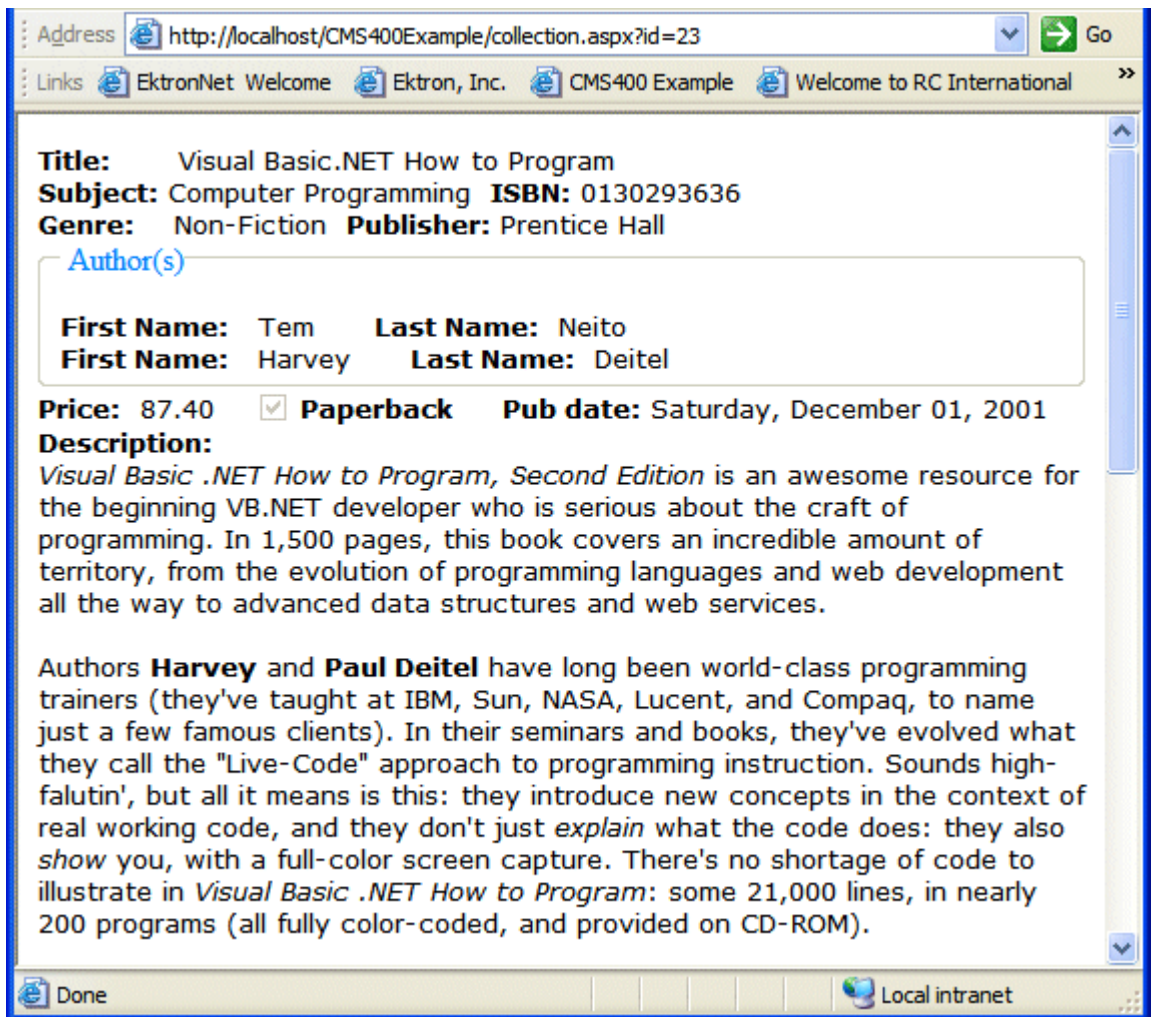
10. A list of books with teasers by that publisher appears.



Notice the second Literal now contains a hyperlink to go back.

11. Click on a book.

12. The XML Content Block for that book appears.



## Turn Off Caching In Netscape Navigator and Mozilla Firefox

When using Netscape Navigator or Mozilla Firefox to perform an index search some parameters become cached. To prevent caching in Netscape Navigator and Mozilla Firefox, add the following line to the Page Load Event in code behind:

```
Response.Cache.SetNoStore()
```

# SiteMap Server Control

The Sitemap server control utilizes the folder breadcrumb information in the Workarea to display a sitemap of your site. By choosing the starting point of the sitemap, the max levels to display and applying a style class, you can customize the sitemap. The sitemap is displayed as indented list when viewed on a Web page.



The contents of the sitemap are defined on the Breadcrumb tab located in the Workarea folder properties. See Also: The Administrator Manual > “Managing Content Folders” > “Breadcrumbs”.

This subsection contains the following topics:

- ["Sitemap Server Control Properties" on page 368](#)
- ["Using the Sitemap Server Control" on page 370](#)

## Sitemap Server Control Properties

The Sitemap server control properties are described in this table.

**NOTE**

The following table only lists Ektron-specific properties. It does not describe native .NET properties such as font, height, width and border style. For documentation of these properties, see VisualStudio help.

Property	Response	Data Type
Authenticated	Indicates if you are logged in to the CMS Explorer and can use it to browse to Content, Collections, etc. See Also: <a href="#">"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 15</a>	String
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a>	Integer
ClassName	The style sheet class name used to format the HTML. Leave blank to use the default. To use a new class, add it to the following file: <code>webroot\siteroot\Workarea\csslib\sitemap.css</code> Then, add the class name to the property.	String
DisplayXslt	Determines how the information on the page is displayed. <b>None</b> -databind only <b>ecmNavigation</b> - lists the title of every content block in the folder See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a> <b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a> <b>Path to Custom Xslt</b> - enter the path to an Xslt that determines the display of the page	String
FlatTopLevel	When set to <b>True</b> , include site nodes from the parent folder but not their items. Set to <b>False</b> to include all nodes and items.	Boolean
FolderID	The folder ID for the starting point of the sitemap. To choose the root folder, enter 0 (zero).	Integer
Hide	Used to hide the Sitemap in design time and run time. <b>True</b> = Hide Sitemap trail <b>False</b> = Show Sitemap trail	Boolean

Property	Response	Data Type
Language	Set a language for viewing content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer
MaxLevel	Set the maximum amount of levels to show. Set it to 0 (zero) for unlimited.	Integer
StartLevel	Set to the starting level of the site map. If set to 0 (zero), starts from the root.	Integer
SuppressWrapperTags	Suppresses the output of the span/div tags around the control. The default is <b>False</b> . <b>True</b> - Suppress wrap tags. <b>False</b> - Allow wrap tags.	Boolean
WrapTag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code. <b>Custom</b> - Allows you to use a custom tag.	String

## Using the Sitemap Server Control

**NOTE** Make sure you have added the sitemap information to your folders' Breadcrumb tab in the CMS400.NET Workarea.

Follow these steps to use the Site server control.

1. Open a Web form for which you want to create a sitemap.
2. Drag and drop the Sitemap server control onto an appropriate location of the Web form.
3. Add the starting folder's ID to the **FolderID** property.
4. Set any of the other properties. See Also: "[Sitemap Server Control Properties](#)" on page 368.
5. Save the Web form.
6. Open a browser.
7. View a Web page with the Sitemap server control in it.



8. The sitemap now appears on your site.



# Dreamweaver Extension Support

Ektron CMS400.NET has a custom Macromedia® Extension Package, which allows you to not only create your templates in Dreamweaver® or UltraDev™, but also to insert the custom functions through Dreamweaver®.

To do this, all you need is Ektron CMS400.NET, Dreamweaver®/UltraDev™, and to keep reading.

Ektron supports Dreamweaver extensions for the following scripting languages:

- ASP.NET - for .NET pages
- ASP - for Active Server Pages
- CFM - for ColdFusion pages
- JSP - Java Server Pages
- PHP - Hypertext Preprocessor

The following sections are contained in this chapter:

- ["Requirements" on page 372](#)
- ["Preparing Dreamweaver for Ektron CMS400.NET Functions" on page 372](#)
- ["Installing the Extensions" on page 373](#)
- ["Setup" on page 377](#)
- ["Editing and Saving Ektron CMS400.NET Content in Dreamweaver" on page 381](#)
- ["Using the Ektron CMS400.NET Extensions" on page 389](#)
- ["Inserting Custom Functions" on page 402](#)

## Requirements

To use a Dreamweaver extension supplied by Ektron, you need to have Dreamweaver 7.0.1 or higher installed, as well as Ektron CMS400.NET.

## Preparing Dreamweaver for Ektron CMS400.NET Functions

To have **Ektron CMS400.NET** function symbols appear in Dreamweaver, while editing a Web page, you must edit two `PreviewModeCustomTags.edml` Dreamweaver files. The steps below explain how to edit the files.

1. In Windows Explorer navigate to the following location

`C:\Program Files\Macromedia\Dreamweaver X\Configuration\Translators\ASP.NET_VB\`

---

**NOTE** In the path above, X represents the version of Dreamweaver you are using. For example, Dreamweaver 8 or Dreamweaver MX.

---

2. Open the `PreviewModeCustomTags.edml` file using Notepad.

3. Search for the following line of code in the file.

```
<searchPattern requiredLocation="openTag"><![CDATA[/runat\s*=\s*"?server"?/i]]></searchPattern>
```

4. Add the following line after it.

```
<searchPattern requiredLocation="tagName"><![CDATA[/^(?!cms:)/i]]></searchPattern>
```

---

**WARNING!** This line must be inserted after the line above. Order is important.

---

5. Save and close the file.

6. Navigate to the following location.

```
C:\Program Files\Macromedia\Dreamweaver X\Configuration\Translators\ASP.NET_Csharp\
```

---

**NOTE** In the path above, X represents the version of Dreamweaver you are using. For example, Dreamweaver 8 or Dreamweaver MX.

---

7. Open the `PreviewModeCustomTags.edml` file using Notepad.

8. Search for the following line of code in the file.

```
<searchPattern requiredLocation="openTag"><![CDATA[/runat\s*=\s*"?server"?/i]]></searchPattern>
```

9. Add the following line after it.

```
<searchPattern requiredLocation="tagName"><![CDATA[/^(?!cms:)/i]]></searchPattern>
```

---

**WARNING!** This line must be inserted after the line above. Order is important.

---

10. Save the file.

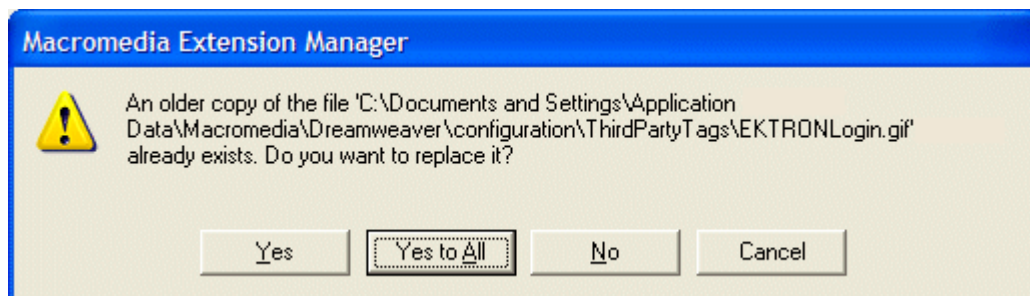
You are now ready to Install the **Ektron CMS400.NET** extension. See Also: ["Installing the Extensions" on page 373](#)

## Installing the Extensions

There are two extensions that can be installed on your server:

- Ektroncms400.mxp - Adds ASP.NET extensions
- C400\_ASP\_PHP\_CF\_JSP.mxp - adds ASP, PHP, CF and JSP extensions

You can have both extensions installed at the same time. The extensions do not affect each other. When you install the second extension, you might receive the following message:



If you see this message or one that say the files are newer, click **Yes to All** to continue. Overwriting the files will not affect the extensions.

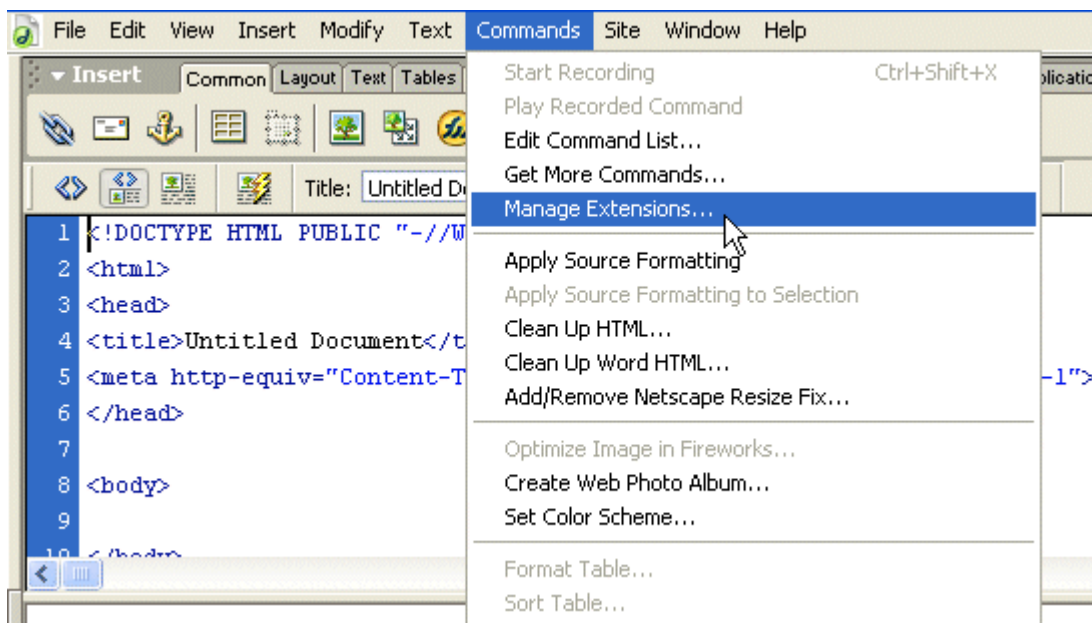
Before you can insert the Ektron CMS400.NET custom functions through Dreamweaver®, you must install at least one of the extensions. Listed below are the steps to install an extension.

**NOTE** For additional information about installing an extension, refer to your Dreamweaver® Manual.

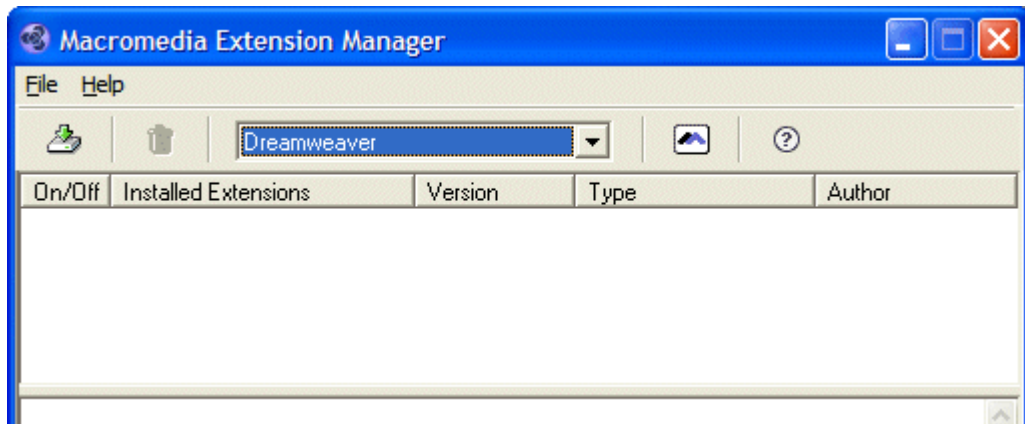
To install an extension, follow these steps.

1. Begin by following the path in Dreamweaver:

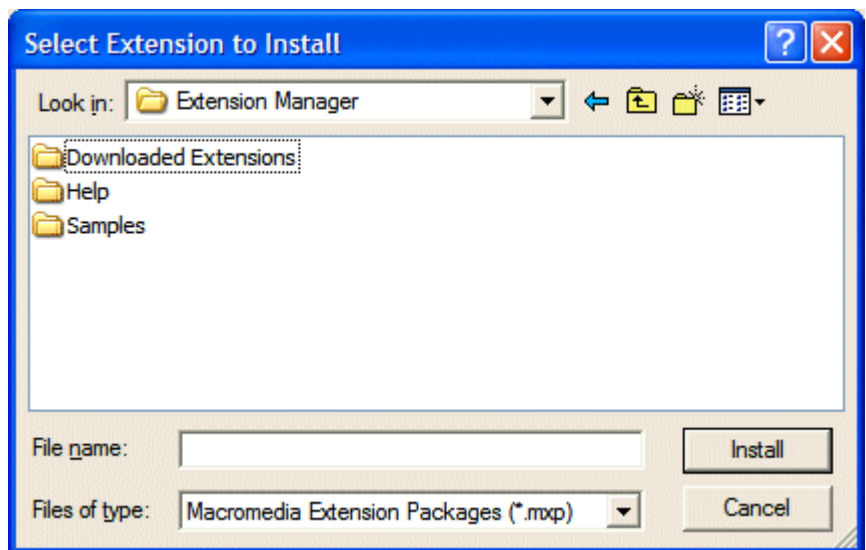
**Commands > Manage Extensions...**



2. The Macromedia Extension Manager is displayed.

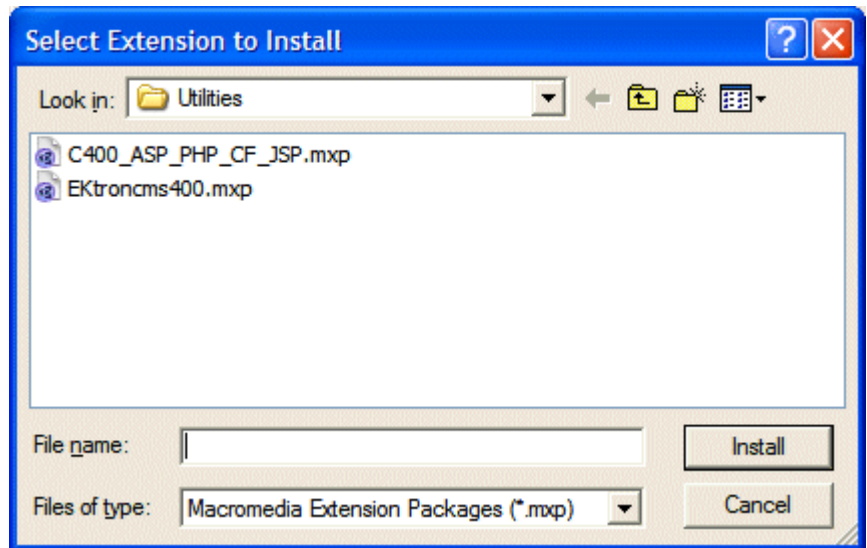


3. From this dialog box, follow the path:  
**File > Install Extension...**
4. The Select Extension to Install window opens.



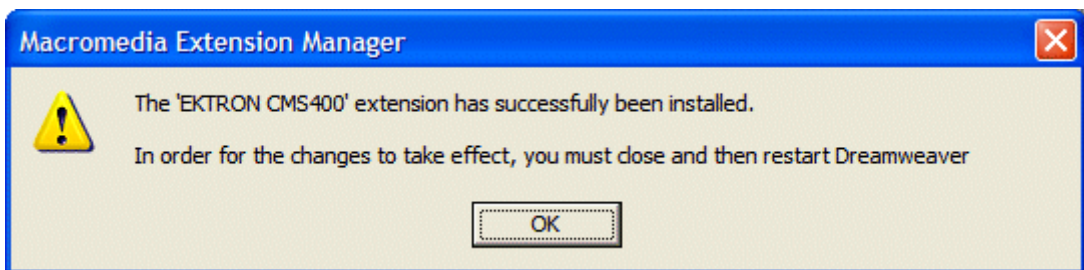
5. Locate the Macromedia® Extension Package (.mxp) file that corresponds to Ektron CMS400.NET. By default, the file gets installed to the following directory:

**C:\Program Files\Ektron\CMS400vXX\Utilities**



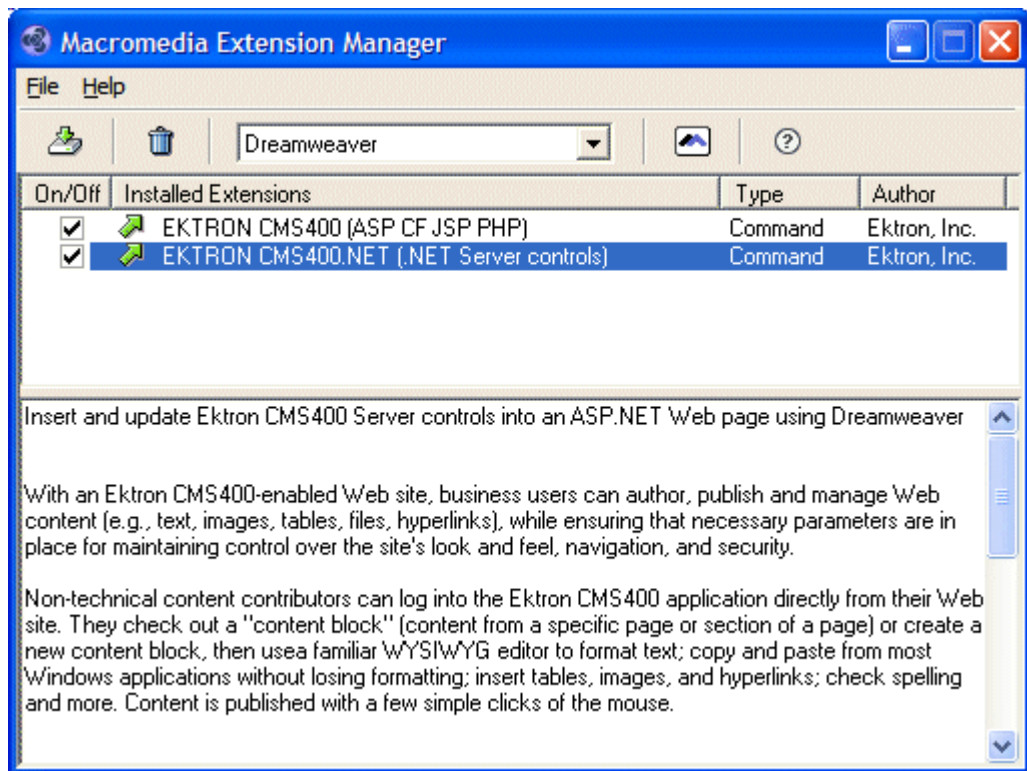
**NOTE** In the folder path, vXX represents the version of **Ektron CMS400.NET** you have installed.

6. After you select the file, click the **Install** button.
7. The installation begins with the Macromedia® Extensions Disclaimer. Click **Accept**.
8. The extension begins to install. Once installed, you see a confirmation box.



9. Click **OK** to continue.

10. The Macromedia Extension Manager is again displayed, but with the added extension.



11. If needed, exit Dreamweaver®, then start it back up.

Now you can use the Ektron CMS400.NET Extension Pack.

Refer to ["Using the Ektron CMS400.NET Extensions" on page 389](#) for more information on using Dreamweaver® to create and edit templates for Ektron CMS400.NET.

## Setup

Before you can use Dreamweaver®/UltraDev™ to create and edit Ektron CMS400.NET templates, you should configure the Ektron CMS400.NET Extension. Listed below are the configurative options for the extension in Dreamweaver:

- Username
- Password
- Domain
- Server File
- Server

- Include
- Check for Include

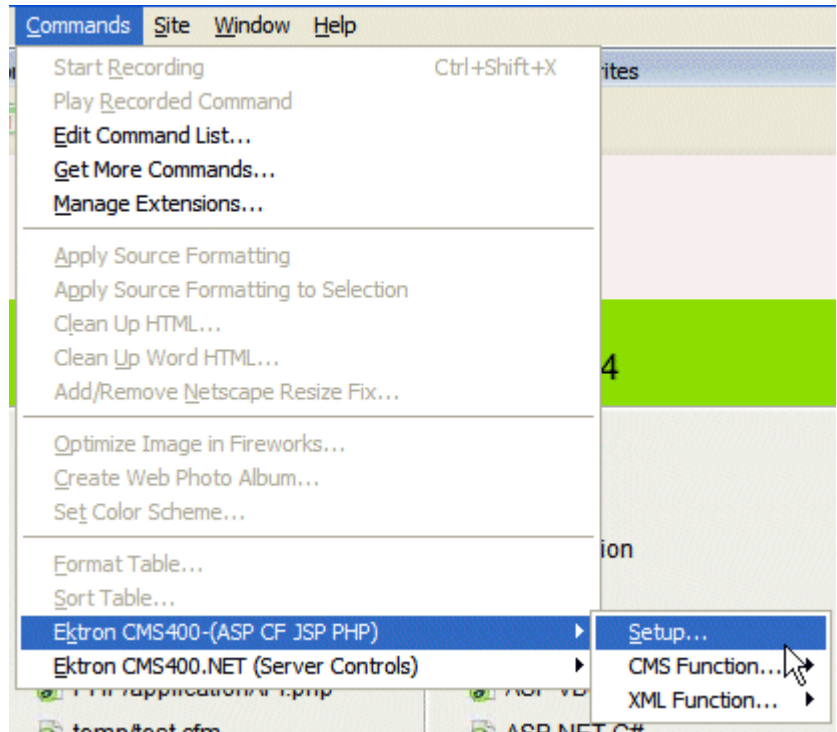
To change these options, follow these steps.

1. If you use ASP, ColdFusion, JSP or PHP, follow this path:

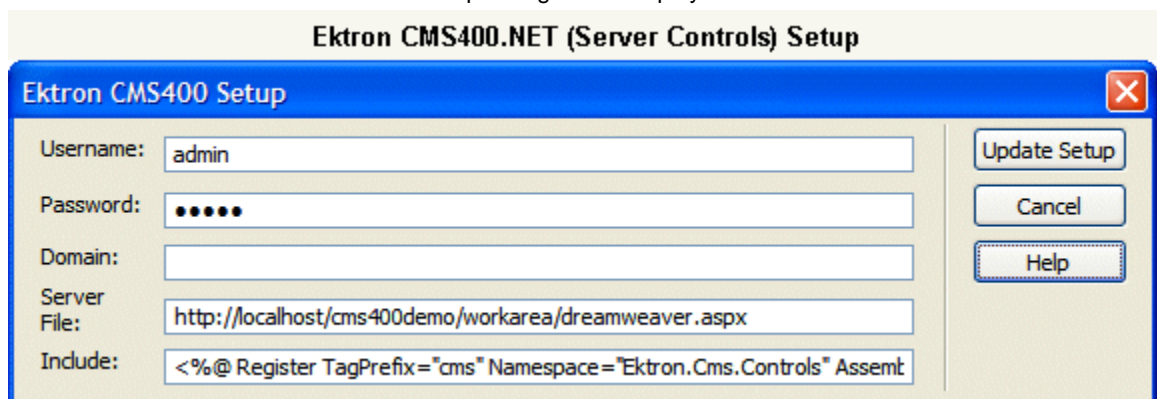
**Commands > Ektron CMS400 - (ASP CF JSP PHP) > Setup...**

If you use ASP.NET, follow this path:

**Commands > Ektron CMS400.NET (Server Controls) > Setup...**



2. The Setup dialog box is displayed on the screen.





**Ektron CMS400 (ASP CF JSP PHP) Setup**

**Ektron CMS400-Platform Setup**

Username:

Password:

Domain:

Server File:

Server: ☒ ASP ☐ JSP ☐ CF ☐ PHP

Include:

☒ Check for include (check is default)

Update Setup  
Cancel  
Help

- Use the following table to help you update your setup information. This table refers to Ektron CMS400.NET (Server Controls) setup. For Ektron CMS400 - (ASP CF JSP PHP), see "[Ektron CMS400 - \(ASP CF JSP PHP\) Dreamweaver Setup Table.](#)" on page 380

Ektron CMS400.NET(Server Control) Dreamweaver Setup Table.

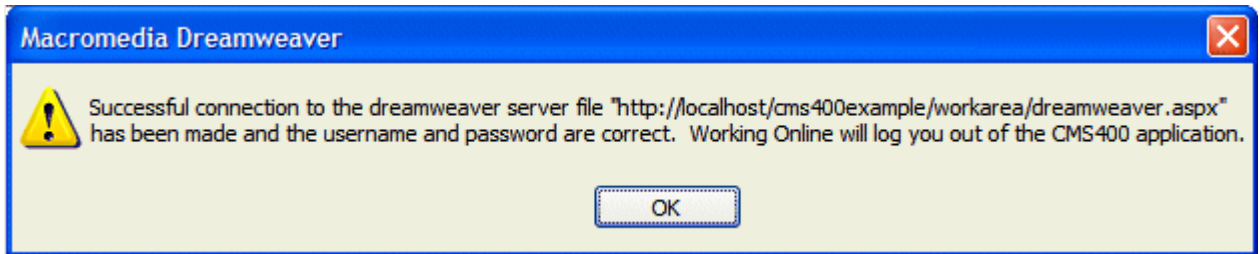
Field	Description
Username	Valid username of a user of your Ektron CMS400.NET Web site. When using the extension, the username is used to log in to the database to retrieve information about content blocks and other Web site information
Password	Enter the password for the user.
Domain	Only required if Active Directory has been enabled for your Web site. Enter the domain corresponding to the username entered.
Server File	Specify the location and file name for the dreamweaver.aspx file needed for the extension to properly work and make connection with your Ektron CMS400.NET database.
Include	Each Ektron CMS400.NET template requires an include file to point to the API with the definitions for the custom functions. Here, specify the include file.

Ektron CMS400 - (ASP CF JSP PHP) Dreamweaver Setup Table.

Field	Description
Username	Valid username of a user of your Ektron CMS400.NET Web site. When using the extension, the username is used to log in to the database to retrieve information about content blocks and other Web site information
Password	Enter the password for the user.
Domain	Only required if Active Directory has been enabled for your Web site. Enter the domain corresponding to the username entered.
Server File	Specify the location and file name for the dreamweaver.aspx file needed for the extension to properly work and make connection with your Ektron CMS400.NET database.
Server	<p>Choose the language of the server.</p> <ul style="list-style-type: none"> <li>• <b>ASP</b> - for Active Server Pages</li> <li>• <b>CFM</b> - for ColdFusion pages</li> <li>• <b>JSP</b> - Java Server Pages</li> <li>• <b>PHP</b> - Hypertext Preprocessor</li> </ul> <hr/> <p>Note: If you want to use a different language, you need to run this setup again and choose a different language.</p> <hr/>
Include	<p>Each template requires an include file to point to the API with the definitions for the custom functions. Here, specify the include file.</p> <ul style="list-style-type: none"> <li>• For ASP use: <code>&lt;!-- #include file="applicationAPI.asp" --&gt;</code></li> <li>• For CFM use: Leave blank</li> <li>• For JSP use: <code>&lt;%@ include file="applicationAPI.jsp" %&gt;</code></li> <li>• For PHP use: <code>&lt;?php include "applicationAPI.php"; ?&gt;</code></li> </ul>
Check for Include	<p>If checked the Include statement is added to your page. Checked is the default.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> - include statement added to your page</p> <p><input type="checkbox"/> <b>Unchecked</b> - include statement is not added to your page</p>

4. Click Update Setup.

5. A message appears.



**NOTE** A message other than this is probably due to an incorrect server file, username, or password. Recheck your settings and try again.

6. Click **OK** to continue.
7. The setup dialog box closes. You are ready to create templates using Macromedia Dreamweaver.

## Editing and Saving Ektron CMS400.NET Content in Dreamweaver

Making the process of setting up and managing your Web site easier, the Ektron CMS400.NET Dreamweaver extensions allow you to perform content related tasks directly from Macromedia Dreamweaver saving you time and effort.

Through Dreamweaver, you can perform the following tasks:

- Add new content blocks to Ektron CMS400.NET
- Edit existing Ektron CMS400.NET content block

The following sections explain how to perform these tasks.

**NOTE** The user specified in the setup information for the extension will only be able to perform tasks they have permissions to in Ektron CMS400.NET.

### Accessing Manage Content Commands...

To access the CMS Manage commands menu, follow this path:

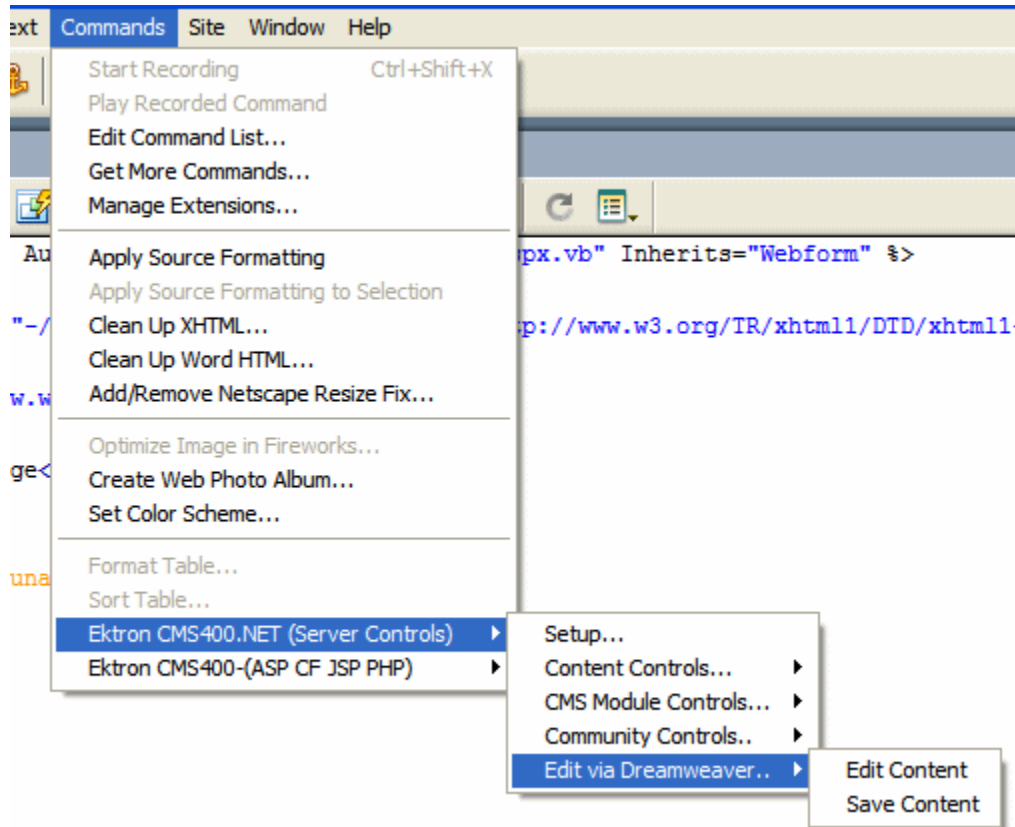
**Commands > Ektron CMS400.NET (Server Controls) > Edit via Dreamweaver...**

Or

**Commands > Ektron CMS400 - (ASP CF JSP PHP) > Edit via Dreamweaver...**

The path you follow depends on which Dreamweaver Extension you installed. If you installed both extensions, you can use either one.

A list of all CMS Manage Content commands that can be performed is displayed.



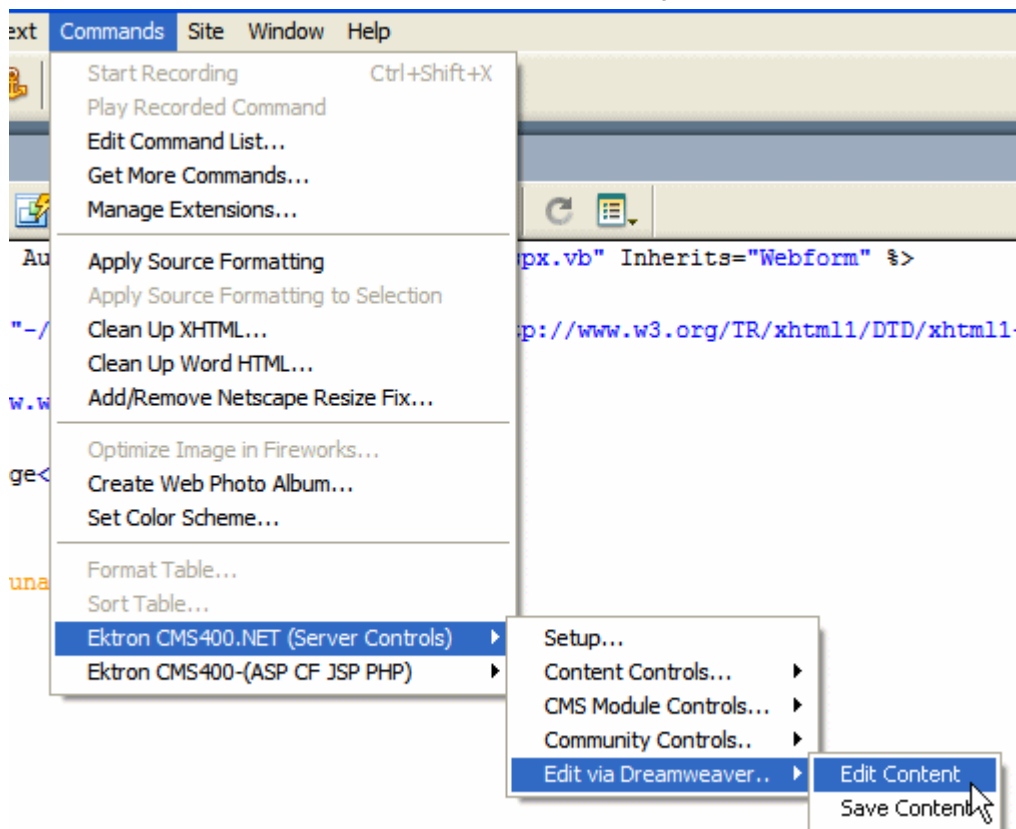
The following sections explain how to use the manage content commands.

## Edit Content

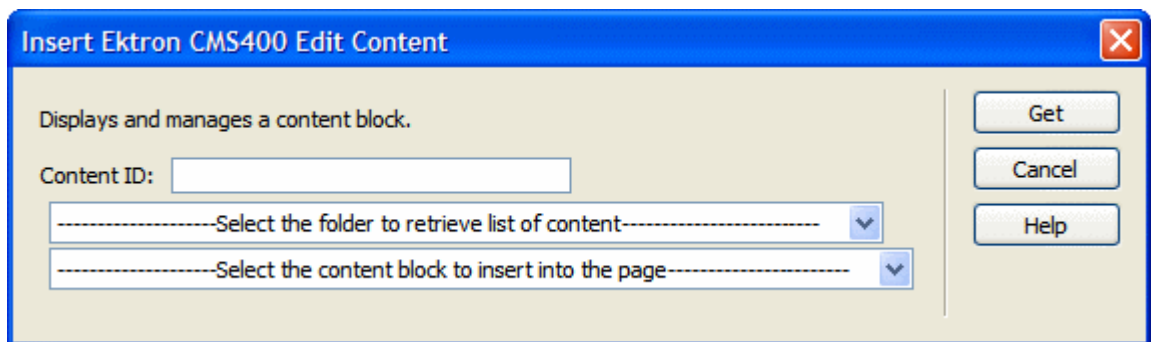
The edit content command allows you to get a content block from the Ektron CMS400.NET Web site, and edit it within Dreamweaver.

To edit a content block within Dreamweaver, perform the following steps:

1. Create a new HTML document in Dreamweaver.
2. In the Ektron CMS400.NET Manage Content menu, click on Edit Content.

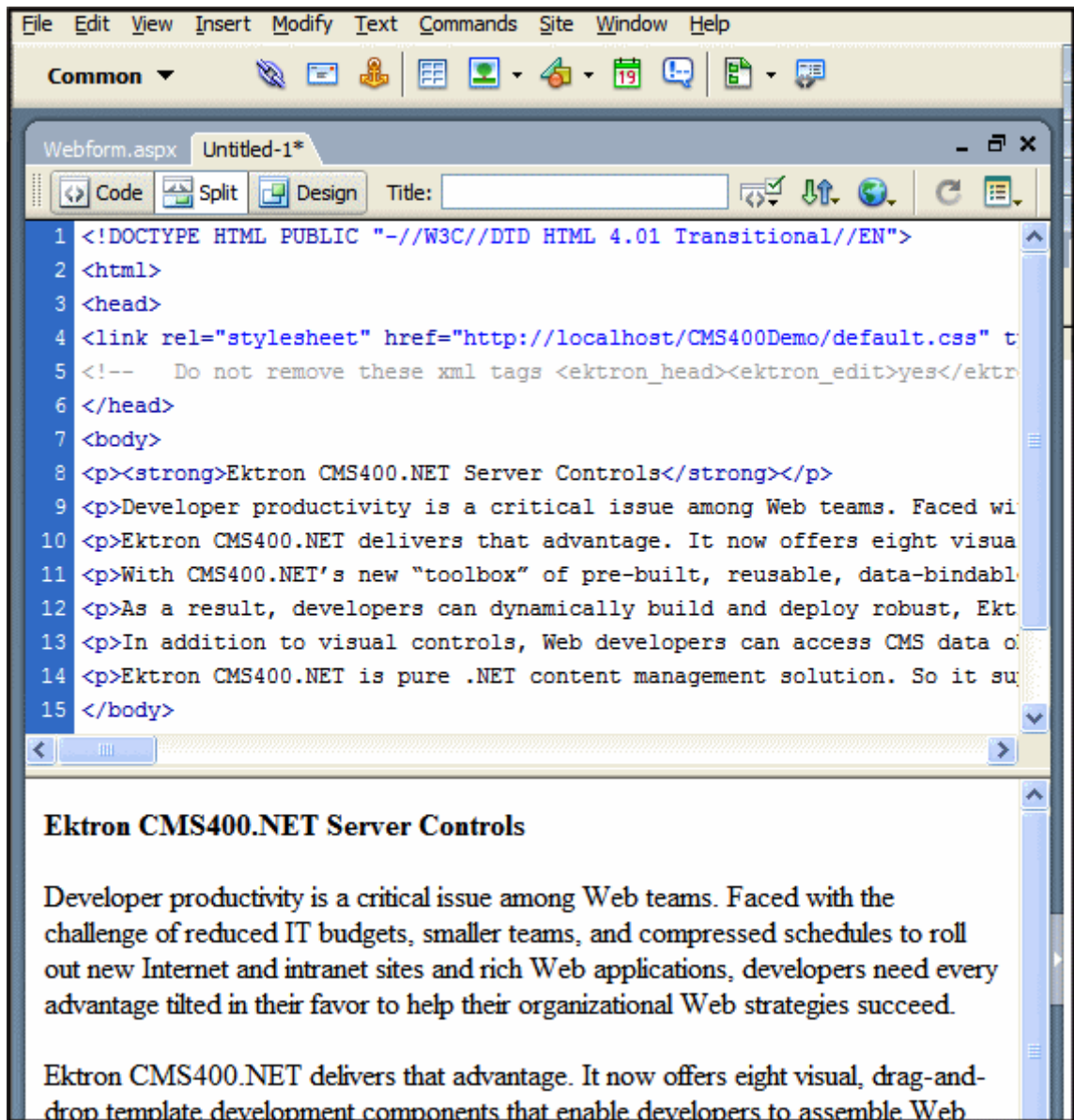


The Insert Content tag dialog box is displayed.



3. Choose the Ektron CMS400.NET content block you would like to edit by:
  - Typing the ID number of the content
  - or
  - Using the drop down lists to locate the content folder and content block

- Click **Get** when you have chosen the content block.  
The content block is inserted into Dreamweaver.



- Make the necessary changes to the content block using Dreamweaver's editing capabilities.  
See ["Saving Existing Ektron CMS400.NET Content"](#) on page 388 for information about saving the content back to Ektron CMS400.NET.

## Save Content

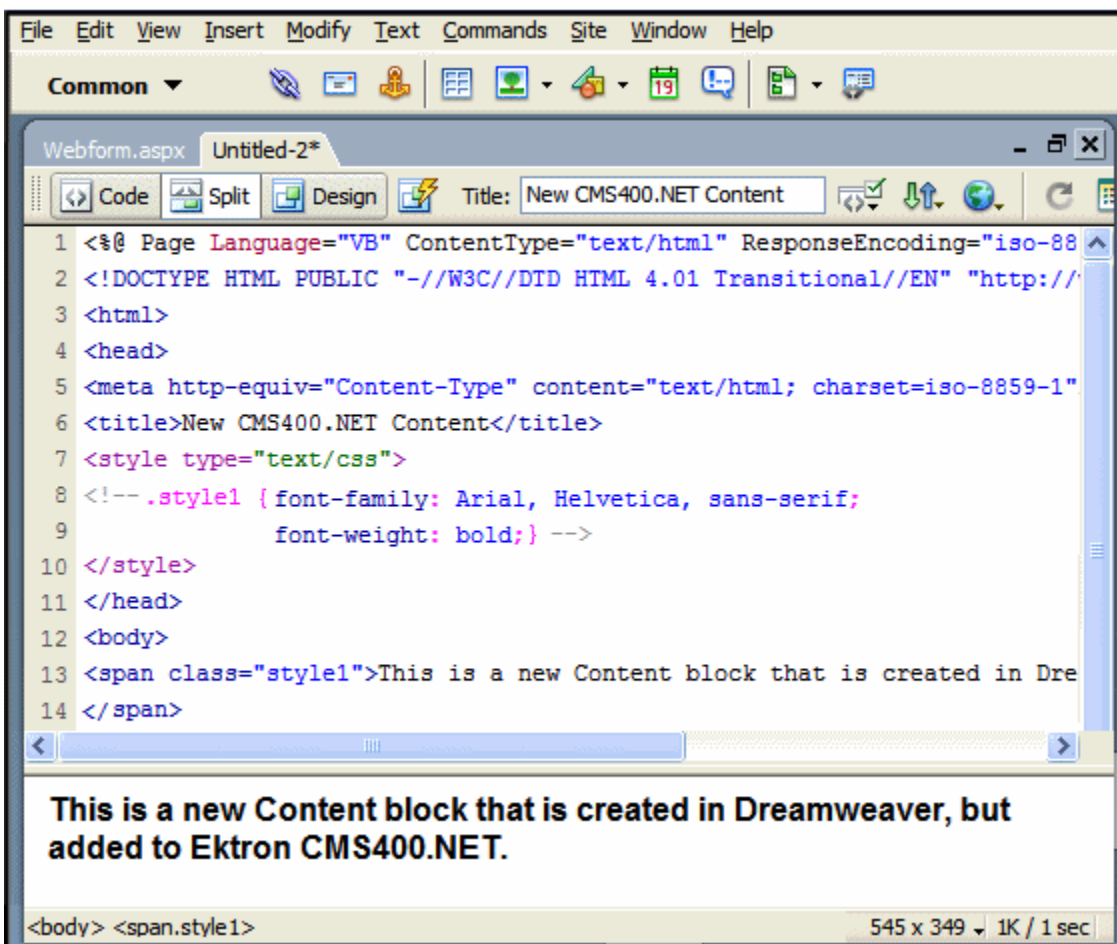
Content that has been created, or edited, in Dreamweaver can be saved to your Ektron CMS400.NET Web site by using the Save Content command.

The following section explains how to save content created, and edited, to Ektron CMS400.NET.

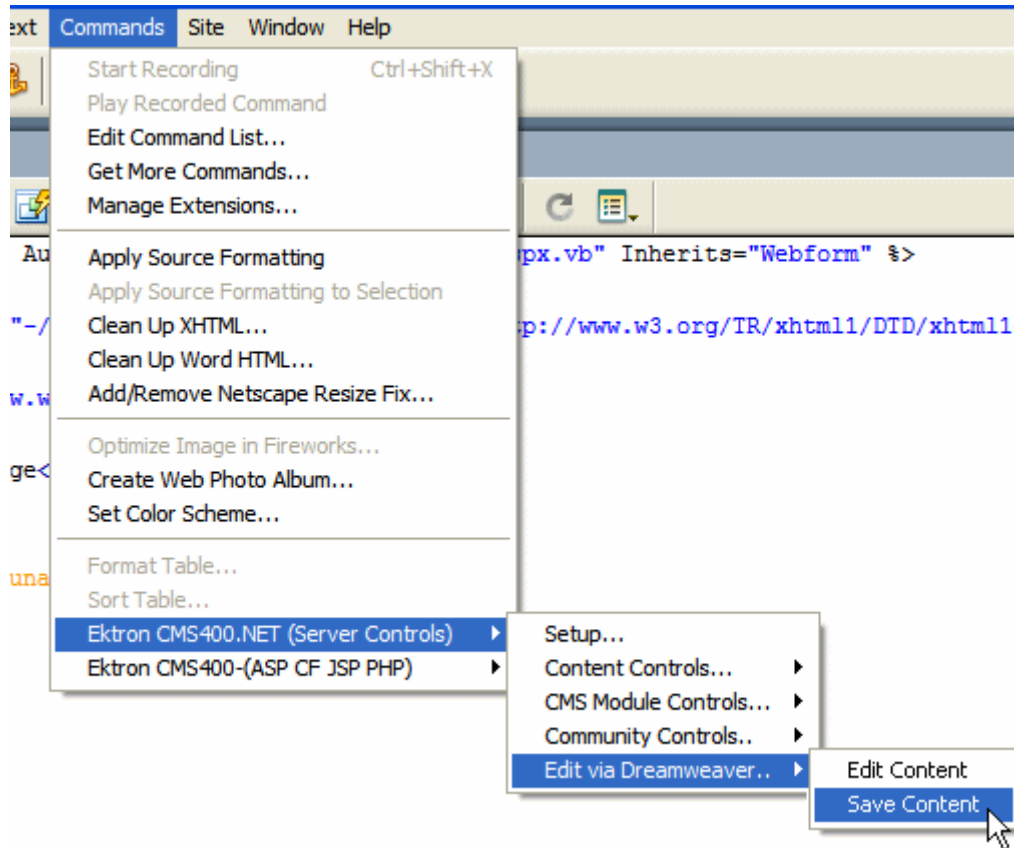
### Saving New Content

To save a content block that was created in Dreamweaver, perform the following steps:

1. If you haven't already, create the new content in Dreamweaver.



2. In the Ektron CMS400.NET Manage Content menu, click on **Save Content**.



The Save Content dialog box is displayed.

A screenshot of the 'Insert Ektron CMS400 Save Content' dialog box. The dialog has a title bar with a close button. It contains four text input fields: 'Title:', 'Comment:', 'Start Date:', and 'End Date:'. Below these is a dropdown menu with the text '-----Select the folder to write content-----'. On the right side of the dialog, there are three buttons: 'Save', 'Cancel', and 'Help'.



3. Using the following table, enter the necessary information to complete the form.

Field	Description
Title	Type a unique title for the content block.
Comment	Enter a comment for the content block.
Start Date	<p>If desired, type in a start date for the content block.</p> <hr/> <p>NOTE: Date format MUST match what is set for the ek_dateFormat element, in the Web.config file, located in the site's webroot. By default, this is <b>dd-mmm-yyyy hh:mm:ss tt</b> (for example, 24-Dec-2005 10:15:00 AM)</p> <hr/>
End Date	<p>If desired, type in an end date for the content block.</p> <hr/> <p>NOTE: Date format MUST match what is set for the ek_dateFormat element, in the Web.config file, located in the site's webroot. By default, this is <b>dd-mmm-yyyy hh:mm:ss tt</b> (for example, 24-Dec-2005 10:15:00 AM)</p> <hr/>
Content Folder	Select the content folder you would like to save the new content block to.

Here is an example of this form filled out.

**Insert Ektron CMS400 Save Content**

Title:

Comment:

Start Date:

End Date:

▼

Save  
Cancel  
Help

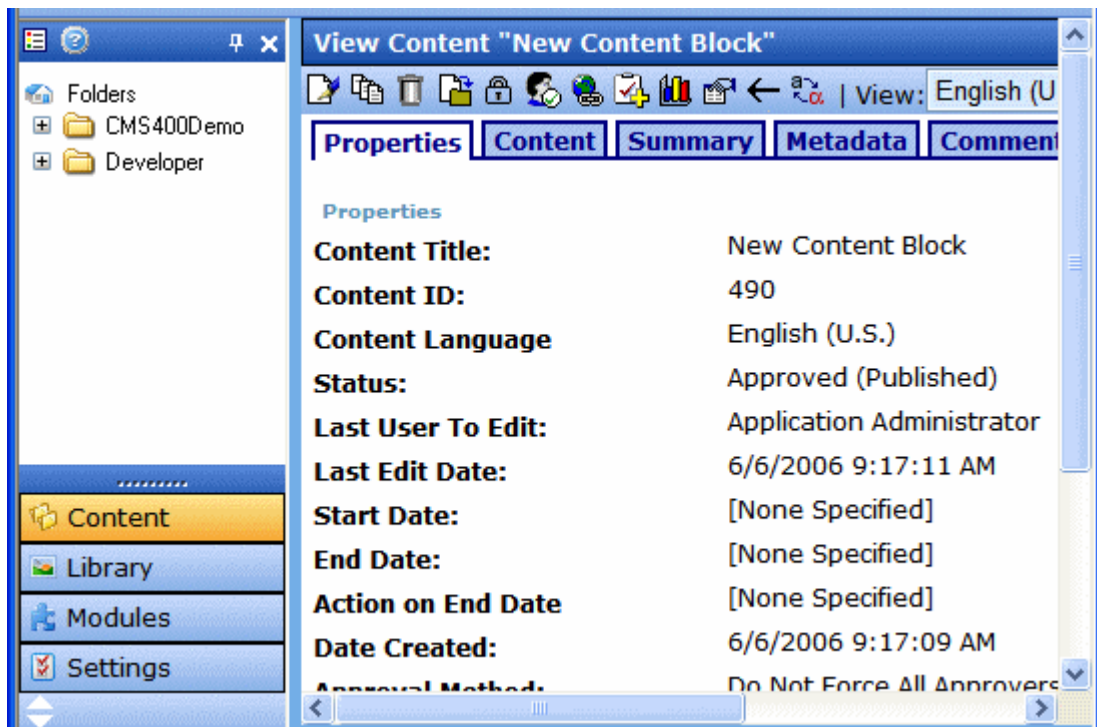
- Click **Save** to save and add the new content block the selected content folder in Ektron CMS400.NET.

The content block is added, and the following confirmation message is displayed.



**NOTE** If you receive a message other than the one displayed above, check your settings and try again.

Here is the new content block in the Ektron CMS400.NET Workarea.



- Once the content has been saved to Ektron CMS400.NET, you can close the page in Dreamweaver.

### ***Saving Existing Ektron CMS400.NET Content***

When Ektron CMS400.NET content has been edited in Dreamweaver, you can use the save content command to save your changes back to Ektron CMS400.NET.

To save existing Ektron CMS400.NET content, perform the following steps.

1. Make changes to an existing Ektron CMS400.NET content block as described in ["Edit Content" on page 382](#).
2. In the Ektron CMS400.NET Manage Content Menu, click on **Save Content**.
3. The Save Ektron Content dialog box is displayed.
4. Use the following table to complete the form.

Field	Description
Title	Editing this field will rename the content block in Ektron CMS400.NET.
Comment	Enter a comment for the content block.
Start Date	<p>If desired, type in a start date for the content block.</p> <hr/> <p><b>NOTE:</b> Date format <b>MUST</b> match what is set for the <code>ek_dateFormat</code> element, in the <code>Web.config</code> file, located in the site's webroot.            By default, this is <b>dd-mmm-yyyy hh:mm:ss tt</b> (for example, 24-Dec-2005 10:15:00 AM)</p> <hr/>
End Date	<p>If desired, type in an end date for the content block.</p> <hr/> <p><b>NOTE:</b> Date format <b>MUST</b> match what is set for the <code>ek_dateFormat</code> element, in the <code>Web.config</code> file, located in the site's webroot.            By default, this is <b>dd-mmm-yyyy hh:mm:ss tt</b> (for example, 24-Dec-2005 10:15:00 AM)</p> <hr/>
Content Folder	Disabled.

5. Click **Save** to save the changes made to the content.  
 The content is saved, and a confirmation message is displayed.

**NOTE** If you receive a message other than one confirming your content has been saved, check your settings and try again.

## Using the Ektron CMS400.NET Extensions










Once the Macromedia® Extension Package has been installed for Dreamweaver, you can create and edit templates for Ektron CMS400.NET.




The following subsections are contained in this section:

- ["Custom Function Symbols" on page 390](#)
- ["Ektron CMS400.NET Command Menus" on page 395](#)


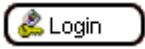

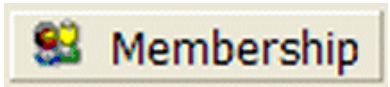



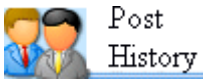
## Custom Function Symbols

During the process of inserting functions, symbols are used to represent the custom display functions. The following table explains each symbol and for which scripting languages the functions are available.

Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
 <b>Active Topics</b>	Active Topics	✓		"ActiveTopics" on page 402
 <b>CMS400 Analytic</b>	Analytics	✓		"Analytics" on page 405
 <b>Asset Control</b>	Asset Control	✓		"AssetControl" on page 408
 <b>CMS400 BLOG</b>	Blog	✓		"Blog" on page 411
 <b>BLOG Archive</b>	BlogArchive	✓		"Blog Archive" on page 415
 <b>BLOG Calendar</b>	Blog Calendar	✓		"Blog Calendar" on page 418
 <b>BLOG Categories</b>	Blog Categories	✓		"Blog Categories" on page 421
 <b>BLOG Entries</b>	Blog Entries	✓		"Blog Entries" on page 424
 <b>BLOG Post</b>	Blog Posts	✓		"Blog Post" on page 428


Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
 <b>BLOG RecentPosts</b>	Blog Recent Posts	✓		"Blog Recent Posts" on page 431
 <b>BLOG Roll</b>	Blog Roll	✓		"Blog Roll" on page 433
 <b>BLOG RSS</b>	Blog RSS	✓		"Blog RSS" on page 436
 <b><i><u>Ektron</u></i> &gt; <i><u>Bread</u></i> &gt; <i><u>Crumb</u></i></b>	BreadCrumb	✓		"BreadCrumb" on page 439
 <b>ektron Calendar</b>	Calendar	✓	✓	"Calendar" on page 445
 <b>ektron Collection</b>	Collection	✓	✓	"Collection" on page 449
 <b>ektron Content Block</b>	Content Block	✓	✓	"Content Block" on page 456
 <b>CMS400 ContentList</b>	Content List	✓		"Content List" on page 460
 <b>Content Rating CMS400</b>	Content Rating	✓		"Content Rating" on page 466
 <b>ektron XML Content Block</b>	Content XSLT Tag		✓	"Content XSLT Tag..." on page 470

Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
	DHTML Menu	✓		"DHTML Menu" on page 471
	Directory - Taxonomy	✓		"Directory-Taxonomy" on page 478
	Discussion Forum	✓		"Discussion Forum" on page 485
	Display by Meta Value	✓		"Display By Meta Value" on page 487
	Dynamic Content Block		✓	"Dynamic Content Block" on page 495
	Folder Bread Crumb	✓		"Folder Bread Crumb" on page 497
	Form Content Block	✓	✓	"Form" on page 499
	Image Control	✓		"ImageControl" on page 503
	Language API	✓		"Language API" on page 506
	Language Select	✓		"Language Select Box" on page 509

Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
	List Summary	✓	✓	"List Summary" on page 511
	Login/Logout button	✓	✓	"Login" on page 520
	Map	✓		"Map Control" on page 525
	Membership	✓		"Membership Control" on page 532
	Metadata	✓	✓	"Metadata for ASP.NET Pages" on page 537 or "Metadata Function for ASP, CF, JSP or PHP Pages" on page 540
	Multipurpose content block.		✓	"Multipurpose Content Block" on page 543
	Poll	✓		"Poll" on page 545
	Post History	✓		"PostHistory" on page 548

Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
	Random Content		✓	"Random Content" on page 551
	Random Summary		✓	"Random Summary" on page 553
	RSS Aggregator	✓		"RSS Aggregator" on page 554
	Old Search	✓	✓	"Old Search" on page 557
	Search display function		✓	"Search Display" on page 568
	Single Summary		✓	"Single Summary" on page 570
	Site Map	✓		"Site Map" on page 573
	Standard Menu	✓		"Standard Menu" on page 580
	Smart Menu	✓		"Smart Menu" on page 576
	Web Search	✓		"Web Search" on page 583



Symbol	Represents	ASP.NET	ASP - CF JSP - PHP	More Information
	XML Index Search	✓		"XML Index Search" on <a href="#">page 589</a>

## Ektron CMS400.NET Command Menus

The following section explains how to locate Ektron CMS400.NET commands in Dreamweaver.

There are two Ektron command menus available. One menu is for ASP.NET pages and the other is used for ASP, ColdFusion, JSP and PHP pages:

- **Commands > Ektron CMS400.NET (Server Controls)**
- **Commands > Ektron CMS400 - (ASP CF JSP PHP)**

To locate a command menu in Dreamweaver, perform the following steps:

1. Open Dreamweaver.
2. In the top menu, follow one of these paths:

- If you are using ASP.NET pages, click

**Commands > Ektron CMS400.NET (Server Controls)**

- If you are using ASP, ColdFusion, JSP or PHP pages, click

**Commands > Ektron CMS400 - (ASP CF JSP PHP)**

For a list of commands in Ektron CMS400.NET (Server Controls), see ["Ektron CMS400.NET \(Server Controls\) Functions" on page 397](#).

For a list of commands in Ektron CMS400 - (ASP CF JSP PHP), see ["Ektron CMS400 - \(ASP CF JSP PHP\) Functions" on page 401](#).

3. The Ektron CMS400.NET commands menu is displayed.

In Ektron CMS400.NET (**Server Controls**) the functions are broken up into the following groups:

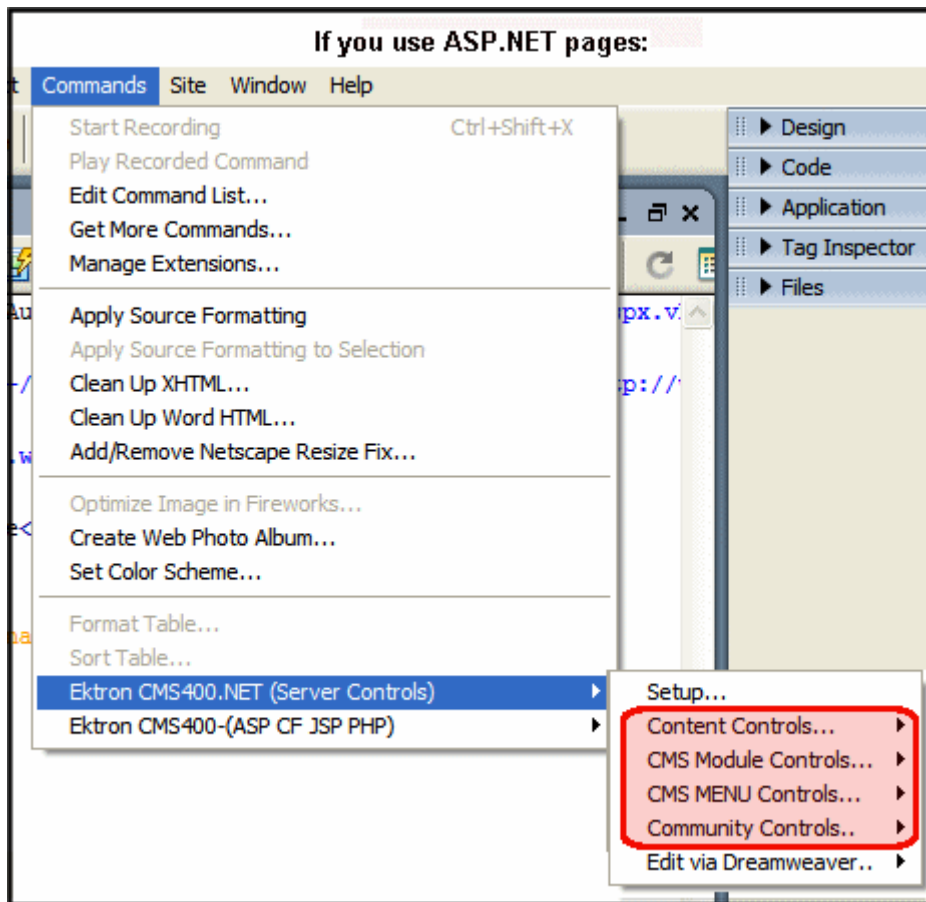
- Content Controls...
- CMS Module Controls...
- Community Controls...

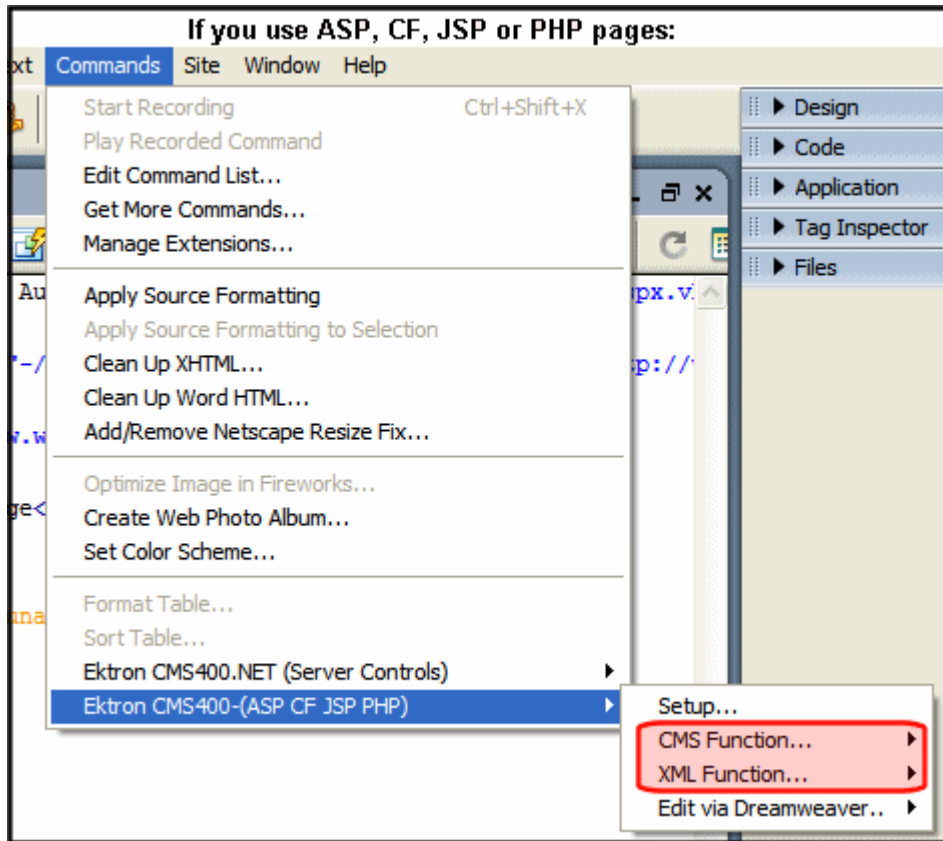
In **Ektron CMS400 - (ASP CF JSP PHP)** the functions are broken up into the following groups:

- CMS Function...
- XML Function...

**NOTE** For a description of the **Setup...** and **Edit via Dreamweaver...** groups, see "Setup" on page 377 and "Editing and Saving Ektron CMS400.NET Content in Dreamweaver" on page 381.

Below are images showing the different views.






4. Click on the command for the task you would like to perform.
- The following tables explain each command. The first table explains the "Ektron CMS400.NET (Server Controls) Functions" on page 397. The second table explains the "Ektron CMS400 - (ASP CF JSP PHP) Functions" on page 401.

#### Ektron CMS400.NET (Server Controls) Functions

Command	Description	More Information
<b>Content Controls...</b>		
Login	Inserts an Ektron CMS400.NET login function into the template.	"Login" on page 520
Content	Inserts an Ektron CMS400.NET content block function into the template.	"Content Block" on page 456
Form	Inserts an Ektron CMS400.NET form content block into the template.	"Form" on page 499

Command	Description	More Information
Poll	Displays a poll or survey created from an Ektron CMS400.NET form on a Web page.	<a href="#">"Poll" on page 545</a>
Collection	Inserts an Ektron CMS400.NET collection function into the template.	<a href="#">"Collection" on page 449</a>
List Summary	Inserts an Ektron CMS400.NET list summary function into the template.	<a href="#">"List Summary" on page 511</a>
Metadata	Inserts an Ektron CMS400.NET metadata function into the template.	<a href="#">"Metadata for ASP.NET Pages" on page 537</a>
Display By Meta Value	Inserts an Ektron CMS400.NET display by meta value block into the template.	<a href="#">"Display By Meta Value" on page 487</a>
Content List	Displays a list of content blocks created from a comma delimited list.	<a href="#">"Content List" on page 460</a>
Language Select	Inserts an Ektron CMS400.NET language select box into the template.	<a href="#">"Language Select Box" on page 509</a>
Language API	Inserts an Ektron CMS400.NET Language API into the template.	<a href="#">"Language API" on page 506</a>
ContentRating	Place a rating scale on any page of your Web site. The scale lets Ektron CMS400.NET collect feedback by giving site visitors the opportunity to rate a particular content item.	<a href="#">"Content Rating" on page 466</a>
WebSearch	The Web Search server control lets you customize the behavior of the search.	<a href="#">"Web Search" on page 583</a>
Old Search	Inserts an Ektron CMS400.NET search function into the template.	<a href="#">"Old Search" on page 557</a>
XML Index Search	Inserts an Ektron CMS400.NET XML Index Search into the template.	<a href="#">"XML Index Search" on page 589</a>
Directory-Taxonomy	Lets you customize the behavior of the Taxonomy feature. You place this control on any Web form to display a taxonomy.	<a href="#">"Directory-Taxonomy" on page 478</a>
<b>CMS Module Controls...</b>		

Command	Description	More Information
Analytics	Track statistics about visits to your Web site.	<a href="#">"Analytics" on page 405</a>
Calendar	Inserts an Ektron CMS400.NET event calendar function into the template.	<a href="#">"Calendar" on page 445</a>
RssAggregator	Aggregates an RSS feed from an RSS Web site.	<a href="#">"RSS Aggregator" on page 554</a>
Map Control	Displays a map that flags locations of interest to your site visitors. Each location is a CMS content item to which map information was added.	<a href="#">"Map Control" on page 525</a>
<b>CMS MENU Controls...</b>		
Folder Bread Crumb	FolderBreadcrumbs show you the path through the sitemap to the current page.	<a href="#">"Folder Bread Crumb" on page 497</a>
BreadCrumb	Creates a breadcrumb trail of where site visitors have been and allows them to navigate back to previous pages.	<a href="#">"BreadCrumb" on page 439</a>
SmartMenu	Displays a menu on a Web page. This menu style is the most robust of the three delivered with <b>Ektron CMS400.NET</b> .	<a href="#">"Smart Menu" on page 576</a>
DHTML Menu	Inserts an Ektron CMS400.NET DHTML menu into the template.	<a href="#">"DHTML Menu" on page 471</a>
Standard Menu	Inserts an Ektron CMS400.NET standard menu into the template.	<a href="#">"Standard Menu" on page 580</a>
Site Map	The Sitemap utilizes the folder breadcrumb information in the Workarea to display a sitemap of your site.	<a href="#">"Site Map" on page 573</a>
<b>Community Controls...</b>		
Membership	Adds a Membership form to your Web form. This allows site visitors to self-register themselves to your site.	<a href="#">"Membership Control" on page 532</a>

Command	Description	More Information
Discussion Forum	Forums provide a discussion board where topics can be discussed on your Web site.	<a href="#">"Discussion Forum" on page 485</a>
ActiveTopics	Displays either the most active topics or the most recent topics.	<a href="#">"ActiveTopics" on page 402</a>
PostHistory	Displays a list of posts for a given users	<a href="#">"PostHistory" on page 548</a>
Blog	The Blog function allows you to quickly add a blog to a Web form. It has all of the items commonly found on a blog page in one server control.	<a href="#">"Blog" on page 411</a>
Blog Calendar	Display a calendar on a Web page and associate it with a blog.	<a href="#">"Blog Calendar" on page 418</a>
Blog Categories	Display the blog categories for a blog on a Web form.	<a href="#">"Blog Categories" on page 421</a>
Blog Roll	Display the blog roll for a blog on a Web form.	<a href="#">"Blog Roll" on page 433</a>
Blog RSS	Display the icon (  ) for the blog's RSS feed on the Web form.	<a href="#">"Blog RSS" on page 436</a>
Blog Entries	Display a list of multiple blog posts on a Web form.	<a href="#">"Blog Entries" on page 424</a>
Blog Recent Posts	Display a list of recent blog posts links on a Web form.	<a href="#">"Blog Recent Posts" on page 431</a>
Blog Post	Display an individual blog post on a Web form.	<a href="#">"Blog Post" on page 428</a>
Blog Archive	Display a list of past months or years that have posts.	<a href="#">"Blog Archive" on page 415</a>
AssetControl	The AssetControl server control, when viewed on a Web form, displays a drag and drop box for users to upload assets to CMS400.NET or update an existing asset.	<a href="#">"AssetControl" on page 408</a>

Command	Description	More Information
ImageControl	Display a CMS400.NET image on a Web page. In addition, if a user has permission to edit the image, he can right click the image and a drag and drop box appears. This box allows a user to overwrite the existing file.	<a href="#">"ImageControl" on page 503</a>

### Ektron CMS400 - (ASP CF JSP PHP) Functions

Command	Description	More Information
<b>Content Controls...</b>		
Login	Inserts an Ektron CMS400.NET login function into the template.	<a href="#">"Login" on page 520</a>
Content	Inserts an Ektron CMS400.NET content block function into the template.	<a href="#">"Content Block" on page 456</a>
Dynamic	Inserts an Ektron CMS400.NET dynamic content block placeholder function into the template.	<a href="#">"Dynamic Content Block" on page 495</a>
Collection	Inserts an Ektron CMS400.NET collection function into the template.	<a href="#">"Collection" on page 449</a>
List Summary	Inserts an Ektron CMS400.NET list summary function into the template.	<a href="#">"List Summary" on page 511</a>
Metadata	Inserts an Ektron CMS400.NET metadata function into the template.	<a href="#">"Metadata for ASP.NET Pages" on page 537</a>
Search	Inserts an Ektron CMS400.NET search function into the template.	<a href="#">"Old Search" on page 557</a>
Search Display Tag	Inserts an Ektron CMS400.NET search display function into the template.	<a href="#">"Search Display" on page 568</a>
Single Summary Tag	Inserts an Ektron CMS400.NET single summary function into the template.	<a href="#">"Single Summary" on page 570</a>
Random Content Tag	Inserts an Ektron CMS400.NET random content function into the template.	<a href="#">"Random Content" on page 551</a>

Command	Description	More Information
Random Summary Tag	Inserts an Ektron CMS400.NET random summary function into the template.	<a href="#">"Random Summary" on page 553</a>
Calendar	Inserts an Ektron CMS400.NET event calendar function into the template.	<a href="#">"Calendar" on page 445</a>
Form	Inserts an Ektron CMS400.NET form content block into the template.	<a href="#">"Form" on page 499</a>
Multipurpose Tag	Insert a content block or form content block into the template.	<a href="#">"Multipurpose Content Block" on page 543</a>
<b>XML Function...</b>		
Content XSLT Tag	<p>Inserts an XML content block with your choice of XSLT in the template.</p> <hr/> <p><b>NOTE:</b> Used to insert an XML content block with an XSLT other than the default applied to it.</p> <hr/>	<a href="#">"Content XSLT Tag..." on page 470</a>

## Inserting Custom Functions

### CMS Function

The CMS Function commands let you insert standard Ektron CMS400.NET display functions into your template.

The following sections explain how to insert each function.

### ActiveTopics

For a description of the ActiveTopics function, see ["ActiveTopics Server Control" on page 155](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Forums**

To insert the ActiveTopics function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **ActiveTopics**.



3. The Insert Ektron CMS400.NET ActiveTopics box appears.

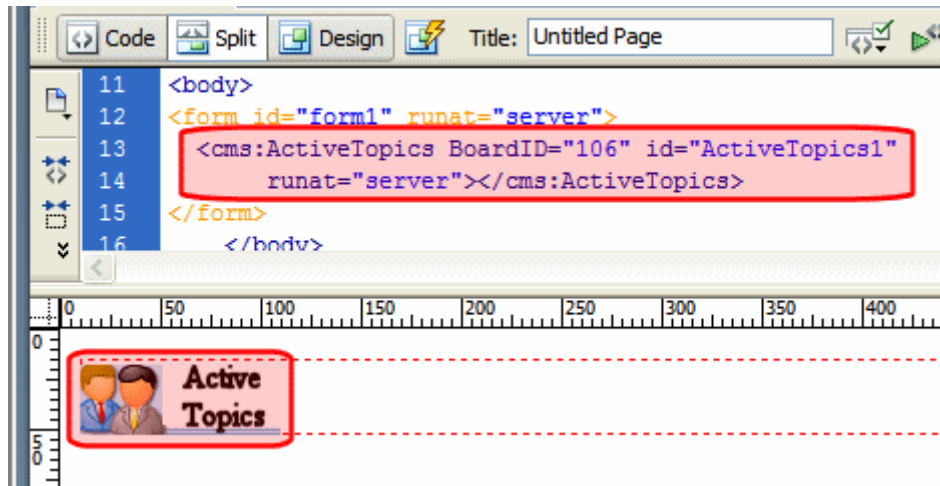
4. Use the following table to assist you with inserting the ActiveTopics function.

Field	Description
Control ID	A unique name for the function you are inserting.
BoardID	The ID of the discussion board. If you do not know the board ID, you can navigate to it using the Select folder drop down box.
DisplayMode	Choose between Active or Recent. In Active mode, the server control displays the most active topics. In Recent mode, the server control displays the topics with the most recent posts. The default is <b>Active</b> .
MaxNumber	The maximum number of topics listed.
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a>

Field	Description
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:  style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

An ActiveTopics icon appears to indicate the location of the active topics list.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:ActiveTopics BoardID="106" id="ActiveTopics1" runat="server"&gt;&lt;/cms:ActiveTopics&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Analytics

For a description of the Analytics function, see ["Content Analytics Server Control" on page 106](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS Module Controls...**

To insert the Analytics function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Analytics**.
3. The Insert Ektron CMS400.NET Analytics box appears.

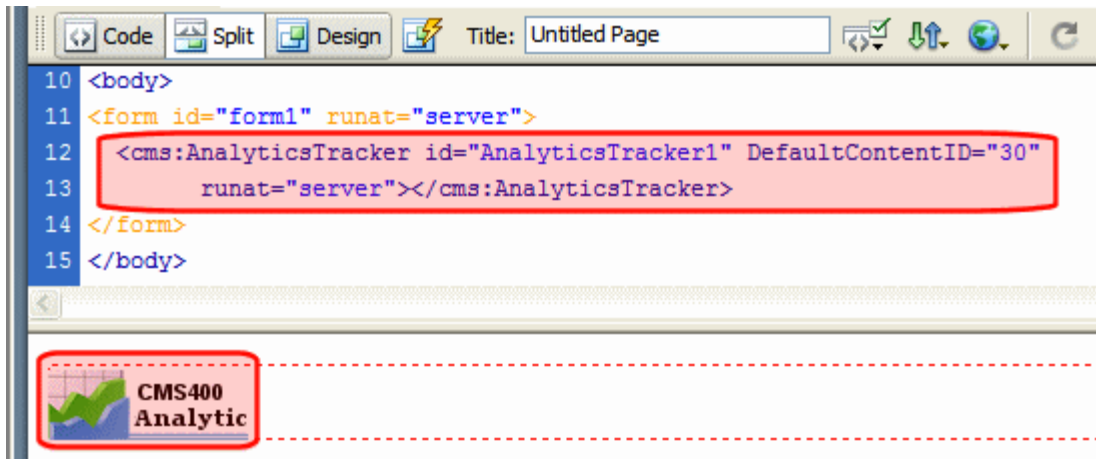
4. Use the following table to assist you with inserting the Analytics function.

Field	Description
Control ID	A unique name for the function you are inserting.
Content ID	Specify the ID number of a content item being tracked by this function. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content drop down boxes.
Dynamic Content Parameter	To make this content analytic control dynamic, select <b>id</b> . When you do, this server control is attached to the content block passed as a URL parameter.
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>

Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

An Analytic icon appears to indicate that content is being tracked. If you are logged in as a CMS user and visit the web page, you see the Analytic information. If you are not logged in, you see nothing.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:AnalyticsTracker id="AnalyticsTracker1" DefaultContentID="30" runat="server"&gt;&lt;/cms:AnalyticsTracker&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## AssetControl

For a description of the AssetControl function, see ["AssetControl Server Control" on page 51](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls...**

To insert the AssetControl function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **AssetControl**.

3. The Insert Ektron CMS400.NET AssetControl box appears.

**Ektron CMS400 DocumentControl**

**AssetControl Control**

ControlID:

FoldeID:

▼

DefaultAssetID:

UploadType:  ▼ DynamicParameter:  ▼

Visible: ☒ Hide: ☐

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:  ▼ SuppressWrapperTags: ☐

Insert Cancel Help

4. Use the following table to assist you with inserting the AssetControl function.

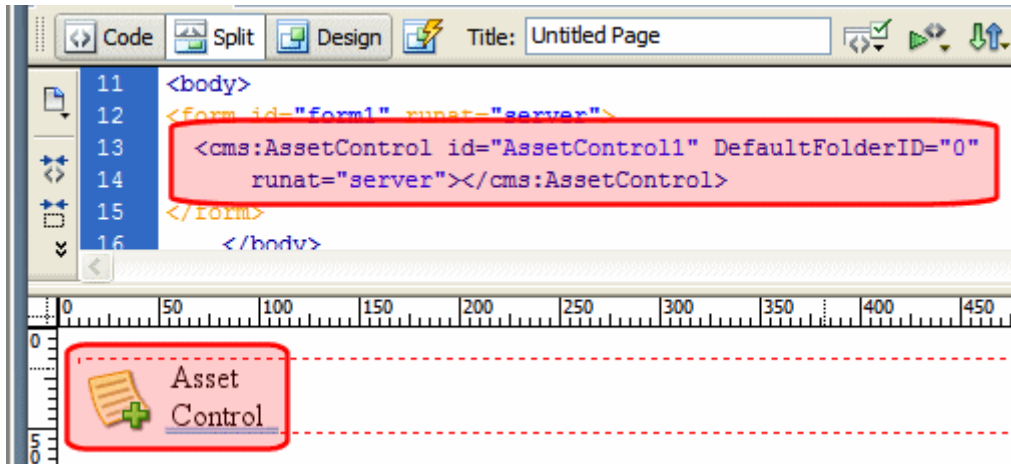
Field	Description
Control ID	A unique name for the function you are inserting.
FolderID	<p>The ID of the folder where assets are added. This property is used when the UploadType property is set to Add. See A/so: "<a href="#">UploadType</a>" on page 409</p> <p>If you do not know the Folder ID, you can navigate to it using the Select Folder drop down box.</p>
DefaultAssetID	<p>The ID of the asset you want to update. This property is use when the UploadType property is set to Update. See A/so: "<a href="#">UploadType</a>" on page 409</p>
UploadType	<p>Select whether the control adds new assets or update existing ones. Select Add to add assets and use the FolderID property. See A/so: "<a href="#">FolderID</a>" on page 409. Select Update to update assets and use the DefaultAssetID. See A/so: "<a href="#">DefaultAssetID</a>" on page 409.</p> <p>The default is <b>None</b>.</p>

Field	Description
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>



5. Click **insert**.

An AssetControl icon appears and indicates the AssetControl's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:AssetControl id="AssetControl1" DefaultFolderID="0" runat="server"&gt;&lt;/cms:AssetControl&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog

For a description of the Blog function, see ["Blog Server Control" on page 54](#) and ["Blog Server Controls" on page 54](#).

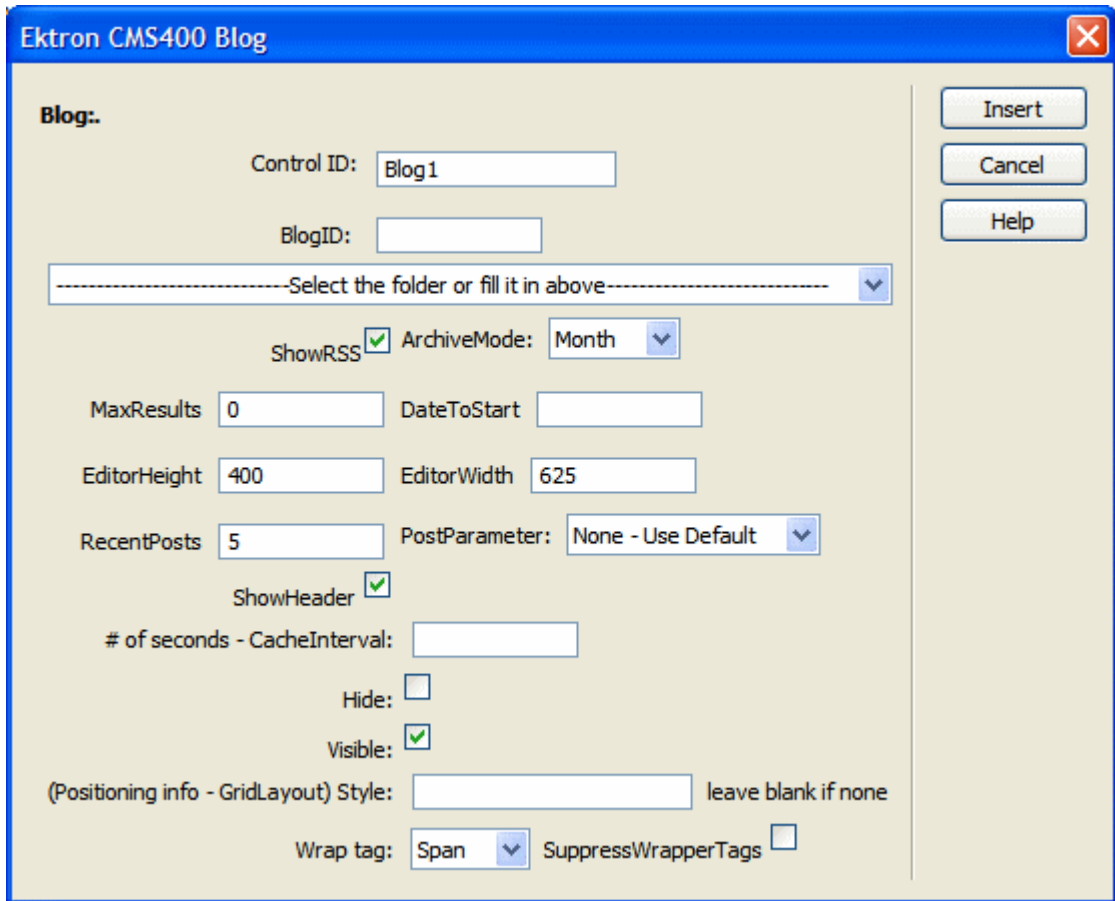
This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**


To insert the Blog function:

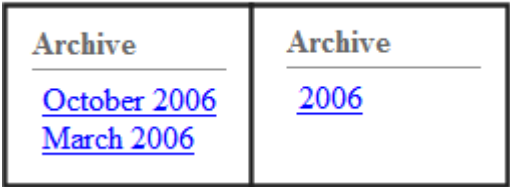
1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog**.

3. The Insert Ektron CMS400.NET Blog box appears.



4. Use the following table to assist you with inserting the Blog function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
ShowRSS	<p>Displays the icon for the RSS feed () when the box is checked. The default is <b>Checked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Display RSS feed</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not display RSS feed</p>

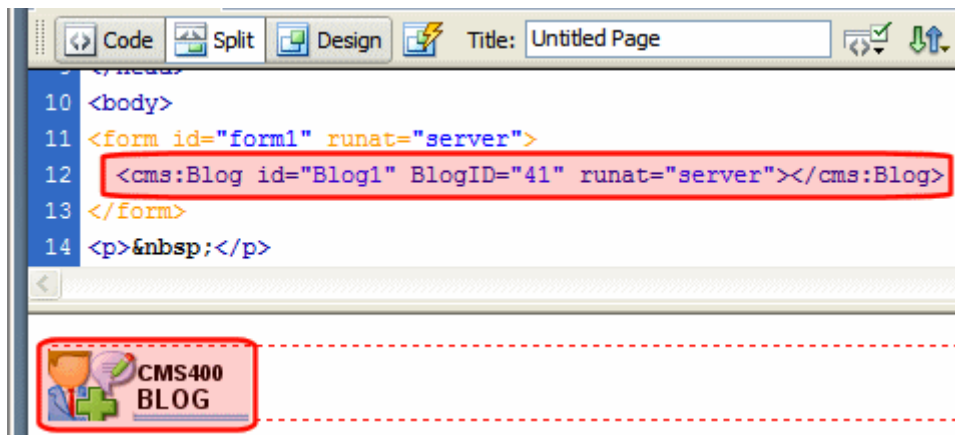
Field	Description
ArchiveMode	<p>Select whether the archive appears in <i>month</i> format or <i>year</i> format. The default is <b>month</b>.</p> <div> <div>Month View</div> <div>Year View</div> </div> 
MaxResults	<p>Set the maximum number of posts to display. If set to 0 (zero), there is no limit. If set to -1, all posts for the day are shown. If set to -2, the all post for the month are shown, If set to -3, the control uses the <b># of Post Visible</b> setting in the Workarea. The default is <b>-3</b>.</p> <p>For information on the Workarea setting <b># of post visible</b>, see the User Manual section “Working With Folders and Content” &gt; “Blogs” &gt; “Blog Properties” &gt; “Blog Folder Properties” &gt; “# Posts Visible”</p>
DateToStart	<p>The date of the last blog entries you want to appear. For example, if you want to display blog entries for January 1, 2006 and before, you enter 1/1/2006. Clicking the dropdown box provides you with a calendar.</p>
EditorHeight	Sets the height of the blog editor in pixels.
EditorWidth	Sets the width of the blog editor in pixels.
RecentPosts	The number of post links contained in the Recent Posts list. The default is 5 (five).
PostParameter	<p>Works like the <code>DynamicParameter</code> for content blocks. When <code>id</code> is selected, this server control passes the blog post ID as a URL parameter.</p> <p>The default setting is <b>id</b>.</p> <p><b>Blank</b> - the list of blog posts is static. The links in the blog posts are inactive.</p> <p><b>id</b> - the id of the blog post is passed to the URL as a parameter.</p> <p><b>None use default</b> - the list of blog posts is static. The links in the blog posts are inactive.</p>

Field	Description
ShowHeader	Shows the title and tagline when this box is checked. The default is Checked. <input checked="" type="checkbox"/> <b>Checked</b> = show title and tagline <input type="checkbox"/> <b>Unchecked</b> = do not show header and tagline
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a>
Visible	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a>
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: <code>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</code>
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.

Field	Description
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

- Click **insert**.

A Blog icon appears and indicates the Blog's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Blog id="Blog1" BlogID="41" runat="server"&gt;&lt;/cms:Blog&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Archive

For a description of the Blog Archive function, see ["BlogArchive Server Control" on page 70](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Archive function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Archive**.
3. The Insert Ektron CMS400.NET Blog Archive box appears.

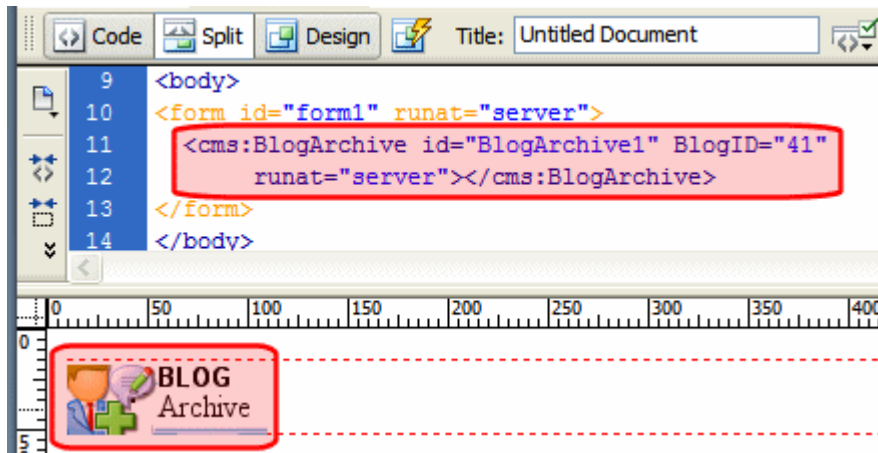
4. Use the following table to assist you with inserting the Blog Archive function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>

Field	Description
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:  style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A BlogArchive icon appears and indicates the Blog Archive's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogArchive id="Blogarchive1" BlogID="41" runat="server"&gt;&lt;/cms:BlogArchive&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Calendar

For a description of the Blog Calendar function, see ["BlogCalendar Server Control" on page 63](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Calendar function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Calendar**.



3. The Insert Ektron CMS400.NET Blog Calendar box appears.

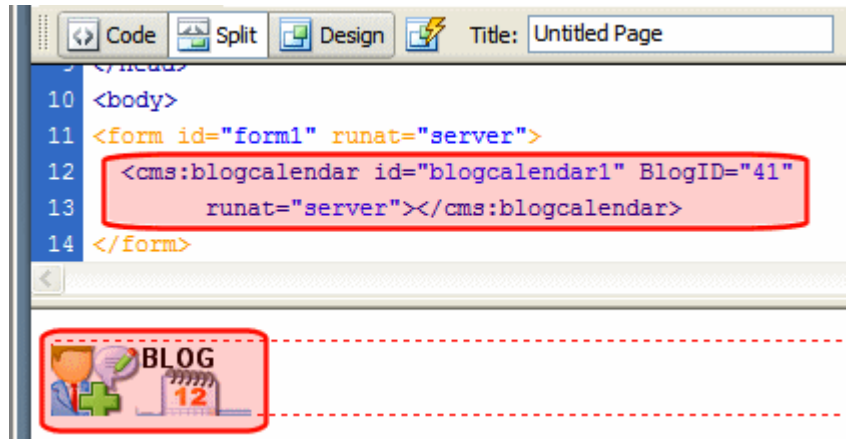
4. Use the following table to assist you with inserting the Blog Calendar function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a>

Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Blog Calendar icon appears and indicates the Blog Calendar's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:blogcalendar id="blogcalendar1" BlogID="41" runat="server"&gt;&lt;/cms:blogcalendar&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Categories

For a description of the Blog Categories function, see "[BlogCategories Server Control](#)" on page 65 and "[Blog Server Controls](#)" on page 54.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Categories function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Categories**.

3. The Insert Ektron CMS400.NET Blog Categories box appears.

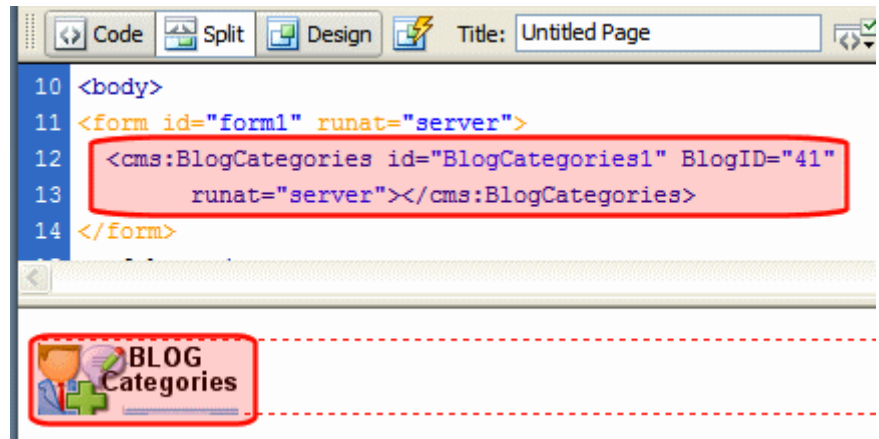
4. Use the following table to assist you with inserting the Blog Categories function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <b>Note:</b> If Hide and Visible are both checked, the property is hidden.

Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Blog Categories icon appears and indicates the Blog Categories' location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogCategories id="BlogCategories1" BlogID="41" runat="server"&gt;&lt;/cms:BlogCategories&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Entries

For a description of the Blog Entries function, see ["BlogEntries Server Control" on page 59](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Entries function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Entries**.

3. The Insert Ektron CMS400.NET Blog Entries box appears.

4. Use the following table to assist you with inserting the Blog Entries function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
DateToStart	The date of the last blog entries you want to appear. For example, if you want to display blog entries for January 1, 2006 and before, you enter 1/1/2006. Clicking the dropdown box provides you with a calendar.
PostParameter	<p>Works like the <code>DynamicParameter</code> for content blocks. When <code>id</code> is selected, this server control passes the blog post ID as a URL parameter. If you do not set this parameter to <code>id</code>, you will not be forwarded to the blog post's page when you click on any links in the post.</p> <p>The default setting is <b>id</b>.</p> <p><b>Blank</b> - the list of blog posts is static. The links in the blog posts are inactive.</p> <p><b>id</b> - the id of the blog post is passed to the URL as a parameter and</p> <p><b>None use default</b> - the list of blog posts is static. The links in the blog posts are inactive.</p>

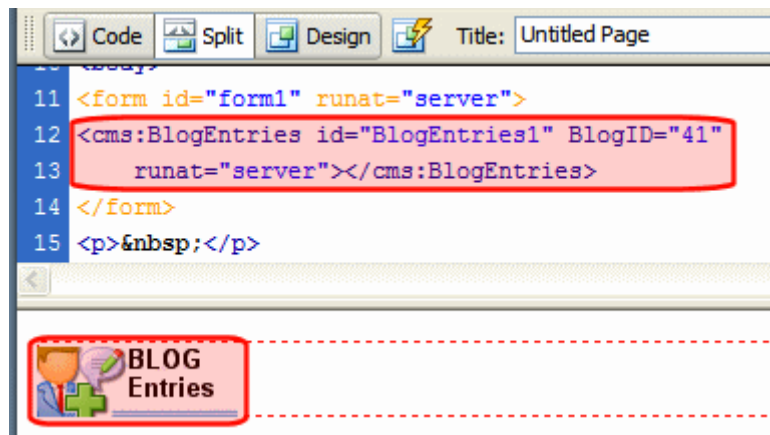
Field	Description
MaxResults	<p>Set the maximum number of posts to display. If set to 0 (zero), there is no limit. If set to -1, all posts for the day are shown. If set to -2, the all post for the month are shown, If set to -3, the control uses the <b># of Post Visible</b> setting in the Workarea. The default is -3.</p> <p>For information on the Workarea setting <b># of post visible</b>, see the User Manual section "Working With Folders and Content" &gt; "Blogs" &gt; "Blog Properties" &gt; "Blog Folder Properties" &gt; "# Posts Visible"</p>
# of seconds - CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a></p>
DisplayXslt	<p>Specify an external XSLT file. See Also: <a href="#">"The OverrideXslt and DisplayXslt Properties" on page 113.</a></p>
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>



Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Blog Entries icon appears and indicates the Blog Entries' location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogEntries id="BlogEntries1" BlogID="41" runat="server"&gt;&lt;/cms:BlogEntries&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.

Platform	Code View
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Post

For a description of the Blog Post function, see ["BlogPost Server Control" on page 61](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Post function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Post**.
3. The Insert Ektron CMS400.NET Blog Post box appears.

**Ektron CMS400 BlogPost**

Blog Post.

Control ID:

DefaultContent ID:

-----Select the category to retrieve list of content-----

-----Select the content block to insert into the page-----

DynamicParameter:

ShowType   DisplayXslt:

Hide: ☐

Visible: ☒

# of seconds - CacheInterval:

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:   SuppressWrapperTags ☐

Insert

Cancel

Help

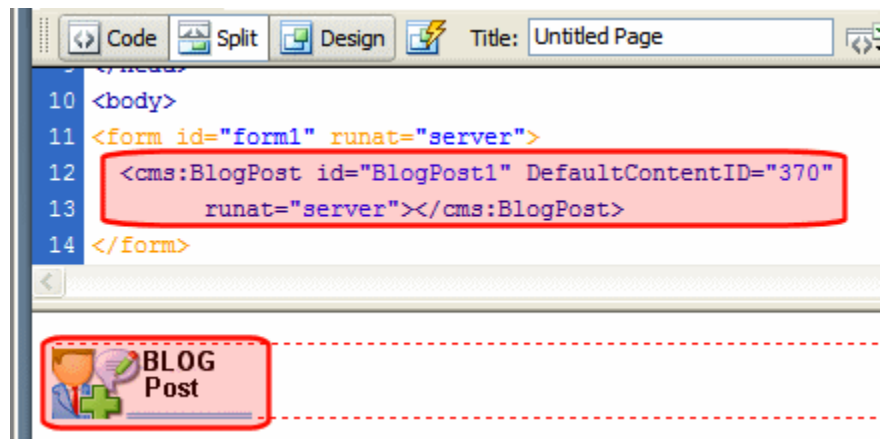
4. Use the following table to assist you with inserting the Blog Post function.

Field	Description
Control ID	A unique name for the function you are inserting.
DefaultContent ID	The ID of a blog post in CMS400.NET. If you do not know the Blog Post's ID, you can navigate to it using the Select Category and Select Content drop down boxes.
DynamicParameter	To make this blog post dynamic, select <b>id</b> . When you do, this server control uses the blog post passed as a URL parameter.
ShowType	Shows a blogpost's content and it's comments or just the blogpost' comments. The default is <b>Content</b> . <b>Content</b> - Show a blogpost's content and its comments. <b>Description</b> - Show a blogpost's comments only.
DisplayXslt	Specify an external XSLT file. See Also: <a href="#">"The OverrideXslt and DisplayXslt Properties"</a> on page 113.
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a>
Visible	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a>
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls"</a> on page 42.

Field	Description
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

5. Click **insert**.

A Blog Post icon appears and indicates the Blog Post's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogPost id="BlogPost1" DefaultContentID="370" runat="server"&gt;&lt;/cms:BlogPost&gt;</code>

Platform	Code View
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Recent Posts

For a description of the Blog Recent Posts function, see "[BlogRecentPosts Server Control](#)" on page 67 and "[Blog Server Controls](#)" on page 54.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Recent Posts function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Recent Posts**.
3. The Insert Ektron CMS400.NET Blog Recent Posts box appears.

**Ektron CMS400 BlogRecentPosts**

Blog Recent Posts

Control ID:

BlogID:

-----Select the folder or fill it in above-----

NumberOfPosts

# of seconds - CacheInterval:

Hide: ☐

Visible: ☒

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag: Span SuppressWrapperTags ☐

Insert  
Cancel  
Help

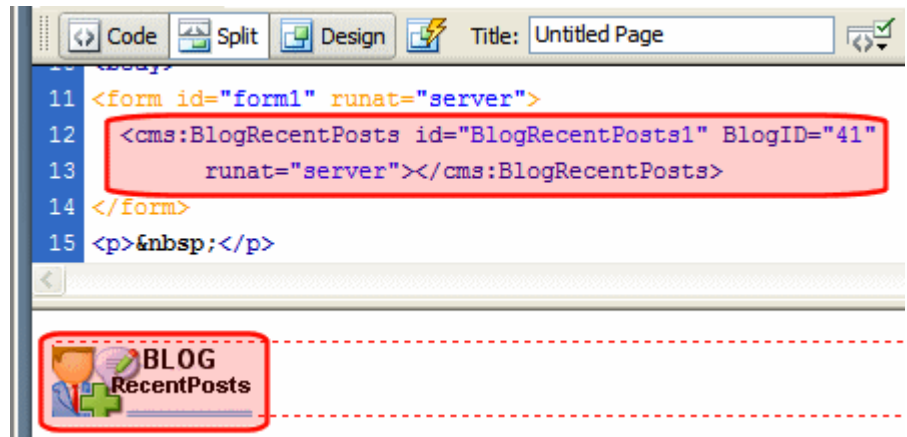
4. Use the following table to assist you with inserting the Blog Recent Posts function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
NumberOfPosts	Sets the number of post links to display.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p><code>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</code></p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.</p>

Field	Description
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Blog Recent Posts icon appears and indicates the Blog Recent Posts' location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogRecentPosts id="BlogRecentPosts1" BlogID="41" runat="server"&gt;&lt;/cms:BlogRecentPosts&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog Roll

For a description of the Blog Roll function, see ["BlogRoll Server Control" on page 64](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog Roll function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog Roll**.
3. The Insert Ektron CMS400.NET Blog Roll box appears.

4. Use the following table to assist you with inserting the Blog Roll function.

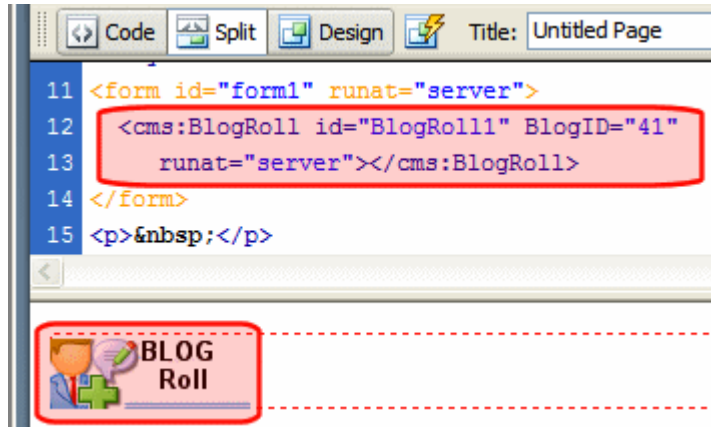
Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>



Field	Description
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **Insert**.

A Blog Roll icon appears and indicates the Blog Roll's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogRoll id="BlogRoll1" BlogID="41" runat="server"&gt;&lt;/cms:BlogRoll&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Blog RSS

For a description of the Blog RSS function, see ["BlogRSS Server Control" on page 68](#) and ["Blog Server Controls" on page 54](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Blogs**

To insert the Blog RSS function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Blog RSS**.

3. The Insert Ektron CMS400.NET Blog RSS box appears.

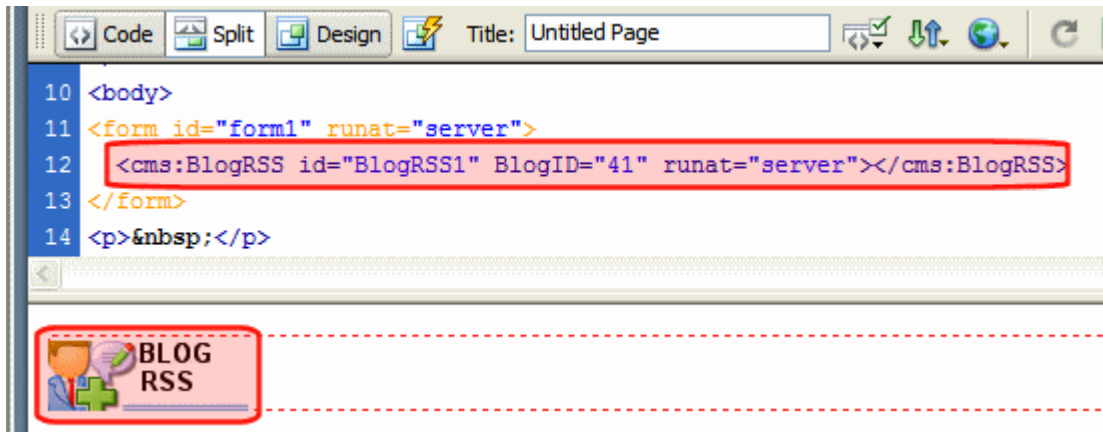
4. Use the following table to assist you with inserting the Blog RSS function.

Field	Description
Control ID	A unique name for the function you are inserting.
Blog ID	The ID of the blog in CMS400.NET. If you do not know the Blog ID, you can choose it from the Select Folder drop down box.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <b>Note:</b> If Hide and Visible are both checked, the property is hidden.

Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **Insert**.

A Blog RSS icon appears and indicates the Blog RSS's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:BlogRSS id="BlogRSS1" BlogID="41" runat="server"&gt;&lt;/cms:BlogRSS&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## BreadCrumb

For a description of the BreadCrumb function, see ["BreadCrumb Server Control" on page 72](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS MENU Controls...**

To insert a BreadCrumb Trail:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **BreadCrumb**.

3. The BreadCrumb dialog box appears.

Ektron CMS400 Bread Crumb

Used to display a Bread Crumb

Control ID:

Meta tag to use - MetadataName:

DisplayTitle:

CurrentPageIndicator:

Content ID:

-----Select the category to retrieve list of content-----

-----Select the content block to insert into the page-----

Dynamic Content Parameter:

IconPath  IconAlt:

LinkTarget   Link Last Item: ☐

Create Hyperlinks ☒ MaxItems:

Separator  Display Style:


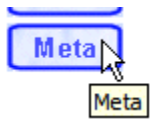
Mode   Hide: ☐

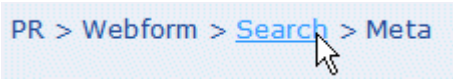
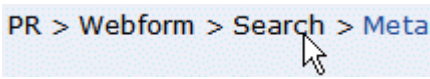

Visible ☒ Wrap tag:   SuppressWrapperTags ☐

(Positioning info - GridLayout) Style:  leave blank if none

4. Use the following table to assist you with inserting a BreadCrumb Trail.

Field	Description
Control ID	A unique name for the function you are inserting.
Meta tag to use - MetadataName	Specify the name of a Metadata Type that you want to associate with the page. For more information, see <a href="#">"BreadCrumb Metadata Type" on page 79</a>

Field	Description
DisplayTitle	<p>Enter text to describe this Web form when it appears in the breadcrumb trail. For example, if the Web form whose properties you are defining is used for all Human Resources pages on your site, enter <b>Human Resources</b>. See <i>Also: "Breadcrumb Properties Apply to Web Forms, Not Web Pages" on page 74</i></p> <p>The default is <b>Title</b>.</p> <p>If you define an image in the <b>IconPath</b> property below, the image appears in the trail, followed by this text.</p>
CurrentPageIndicator	<p>If desired, enter one or more symbols or characters that identify the current page in the breadcrumb trail. See example below.</p> <p style="text-align: center;">Dev &gt; WebForm &gt; pr &gt; FAQ &gt; Meta </p> <p>These characters appear <i>after</i> the image or title that identifies the current page in the breadcrumb trail.</p>
Content ID	Gets or Sets the ContentID for the Breadcrumb display title. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content drop down boxes.
Dynamic Content Parameter	Gets or sets the QueryString parameter to read a content ID dynamically.
IconPath	<p>If you want the breadcrumb trail to display an image to identify this Web form, enter the path to the image.</p> <hr/> <p><b>Important:</b> The image location must be relative to the Web root.</p> <hr/> <p>For example: \CMS400Min\Workarea\Images\bc_meta_icon.gif</p>
IconAlt	<p>If you define an image in the <b>IconPath</b> property, enter any "Alt" text that should appear when a site visitor hovers the cursor over that image. Here is an example.</p> <p style="text-align: center;"></p>

Field	Description
LinkTarget	<p>Determines the type of window that appears on this Web form when a user clicks an item in the breadcrumb trail. The default is <b>_self</b>.</p> <p><b>_Self</b> - opens in same window  <b>_Top</b> - opens in parent window  <b>_Blank</b> - opens in new window  <b>_Parent</b> - opens in the parent frame</p>
Link Last Item	<p>Use this property to determine whether the last breadcrumb item appears as a hyperlink on this Web form. If this property box is checked, and a user clicks the item, the current page reappears.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = last item is a hyperlink</p> <p><input type="checkbox"/> <b>Unchecked</b> = last item is an image and/or text only; the user cannot click on it</p>
Create Hyperlinks	<p>Unchecking this box makes the breadcrumb trail appear as non-hyperlinked plain text. The default is Checked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = breadcrumb trail is hyperlinked</p>  <p><input type="checkbox"/> <b>Unchecked</b> = breadcrumb trail is plain text</p> 
MaxItems	<p>Enter the maximum number of items in the breadcrumb trail on this Web form. The default is <b>5</b>.</p> <p>If you set a value of 1 or greater and the user navigates beyond that number of pages, only the most recent pages appear. The older pages disappear from the trail.</p> <p>To allow an unlimited number of breadcrumb trail items on this Web form, enter zero (<b>0</b>).</p>
Separator	<p>Enter one or more characters to separate the items in a breadcrumb trail on this Web form.</p> <p>The default character is the greater than sign (&gt;).</p> 

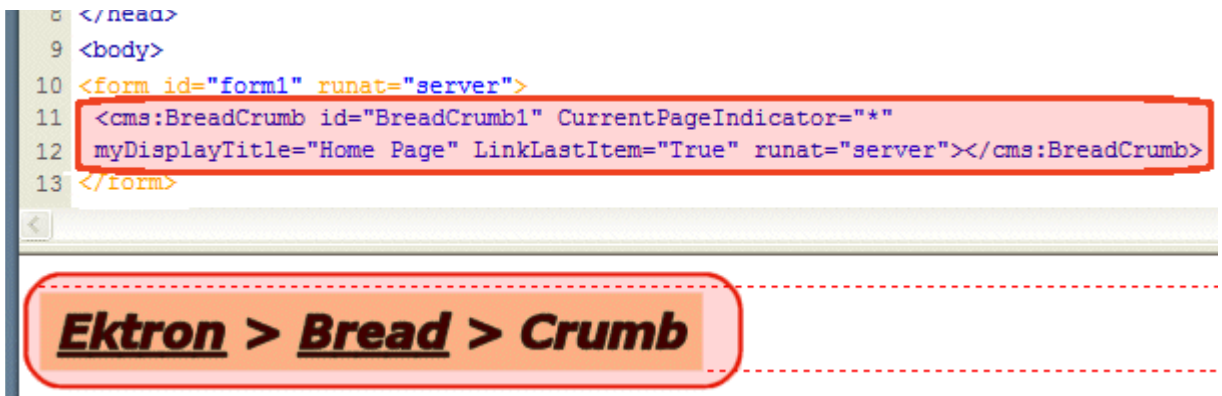


Field	Description
DisplayStyle	<p>Indicate how to display the breadcrumb trail: horizontally or vertically. The default is Horizontal.</p> <p>This is an example of <b>Horizontal</b>.</p> <p><a href="#">Dev</a> &gt; <a href="#">WebForm</a> &gt; <a href="#">pr</a> &gt; <a href="#">FAQ</a> &gt; <a href="#">Meta*</a></p> <p>This is an example of <b>Vertical</b>.</p> <p>&gt; <a href="#">Dev</a> &gt; <a href="#">WebForm</a> &gt; <a href="#">pr</a> &gt; <a href="#">FAQ</a> &gt; <a href="#">Meta*</a></p>
Mode	<p>Allows you to make the breadcrumb trail appear as non-hyperlinked plain text. The default is <b>Normal</b>.</p> <p><b>Normal</b> = breadcrumb trail is hyperlinked</p> <p><a href="#">PR</a> &gt; <a href="#">Webform</a> &gt; <a href="#">Search</a> &gt; <a href="#">Meta</a></p> <p><b>DisplayOnly</b> = breadcrumb trail is plain text</p> <p><a href="#">PR</a> &gt; <a href="#">Webform</a> &gt; <a href="#">Search</a> &gt; <a href="#">Meta</a></p>
Hide	<p>Used to hide a BreadCrumb in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide BreadCrumb</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show BreadCrumb</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>

Field	Description
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"

5. When completed, click **Insert**.

A graphic defines where the BreadCrumb Trail appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:BreadCrumb id="BreadCrumb1" CurrentPageIndicator="" * myDisplayTitle="Home Page" LinkLastItem="True" runat="server"></cms:BreadCrumb>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.

Platform	Code View
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Calendar

For a description of the Calendar function, see ["Calendar Server Control" on page 88](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- **Commands > Ektron CMS400.NET (Server Controls) > CMS Modules Controls...**

To insert an Calendar function:

---

**NOTE** Inserting a calendar function does not add a reference to a calendar style sheet. This step must be performed manually.

---

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click on **Calendar**

3. The Insert Calendar box is displayed.

**IMPORTANT!**

When inserting an event calendar using ASP, CF, JSP or PHP, the dialog box that appears only allows you to set the Calendar ID or choose one from the list.

4. Use the following table to assist you with inserting a calendar.

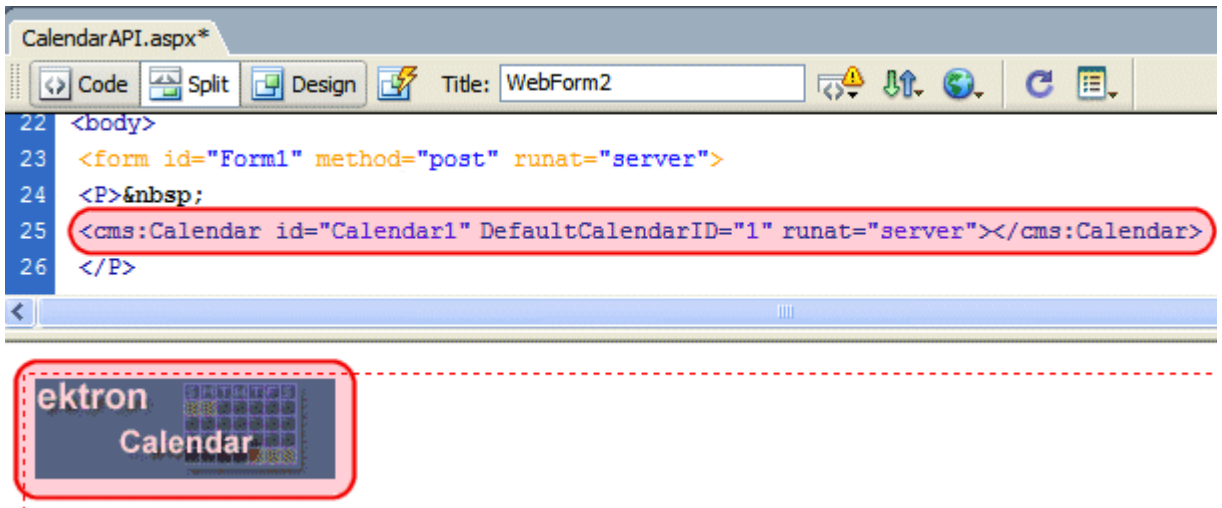
Field	Description
Control ID <u>Used in ASP.NET only.</u>	A unique name for the function you are inserting.

Field	Description
Content ID	Type in the Content ID of the calendar that you want to insert. If you do not know the Content ID, you can navigate to it using the Select Calendar drop down box.
Select Calendar	The drop-down box is populated with a list of all the calendar titles. When a title is selected, the content ID textbox is filled with that content block's id number. If you insert the calendar ID, you do not need to select a calendar.
Dynamic Content Parameter  <u>Used in ASP.NET only.</u>	To make this calendar dynamic, select <b>calendar_id</b> . When you do, this server control uses the calendar passed as a URL parameter.
DisplayType  <u>Used in ASP.NET only.</u>	Specify the Calendar view type. The default is <b>Month</b> . <ul style="list-style-type: none"> <li>• <b>Month</b> - A 30 day calendar month view.</li> <li>• <b>Day</b> - Single day view.</li> <li>• <b>Inline</b> - Display events that occur from the start date of the calendar through the end date.</li> <li>• <b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.</li> </ul>
ShowXML  <u>Used in ASP.NET only.</u>	At runtime, output the raw XML data instead of the calendar. With this XML data, you can apply your own XSL to create your own custom display formats. Default is <b>Unchecked</b> .  <input checked="" type="checkbox"/> <b>Checked</b> = Show raw XML data <input type="checkbox"/> <b>Unchecked</b> = Show Calendar
EndDate  <u>Used in ASP.NET only.</u>	Specify the end date the calendar displays. Used in conjunction with the StartDate property. For example, you have a calendar that shows only classes for the next three month semester.
StartDate  <u>Used in ASP.NET only.</u>	Specify the start date the calendar displays. Used in conjunction with the EndDate property. For example, you have a calendar that shows only classes for the next three month semester.

Field	Description
Hide <hr/> <i>Used in ASP.NET only.</i>	Used to hide a calendar in design time and run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide calendar <input type="checkbox"/> <b>Unchecked</b> = Show calendar <hr/> <i>Note: If Hide and Visible are both checked, the property is hidden.</i>
Visible <hr/> <i>Used in ASP.NET only.</i>	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> <i>Note: If Hide and Visible are both checked, the property is hidden.</i>
Style (Positioning Info - GridLayout) <hr/> <i>Used in ASP.NET only.</i>	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag <hr/> <i>Used in ASP.NET only.</i>	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

- When completed, click **insert**.

A graphic defines where the calendar appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:Calendar id="Calendar1" DefaultCalendarID="1" runat="server"&gt;&lt;/cms:Calendar&gt;</code>
ASP	<code>&lt;% ecmEvtCalendar (1) %&gt;</code>
ColdFusion	<code>&lt;cfmodule Calendar_id="1" template="#request.ecm.AppPath#ecmEvtCalendar.cfm"&gt;</code>
JSP	<code>&lt;%= ecmEvtCalendar(1) %&gt;</code>
PHP	<code>&lt;?php ecmEvtCalendar (1); ?&gt;</code>

## Collection

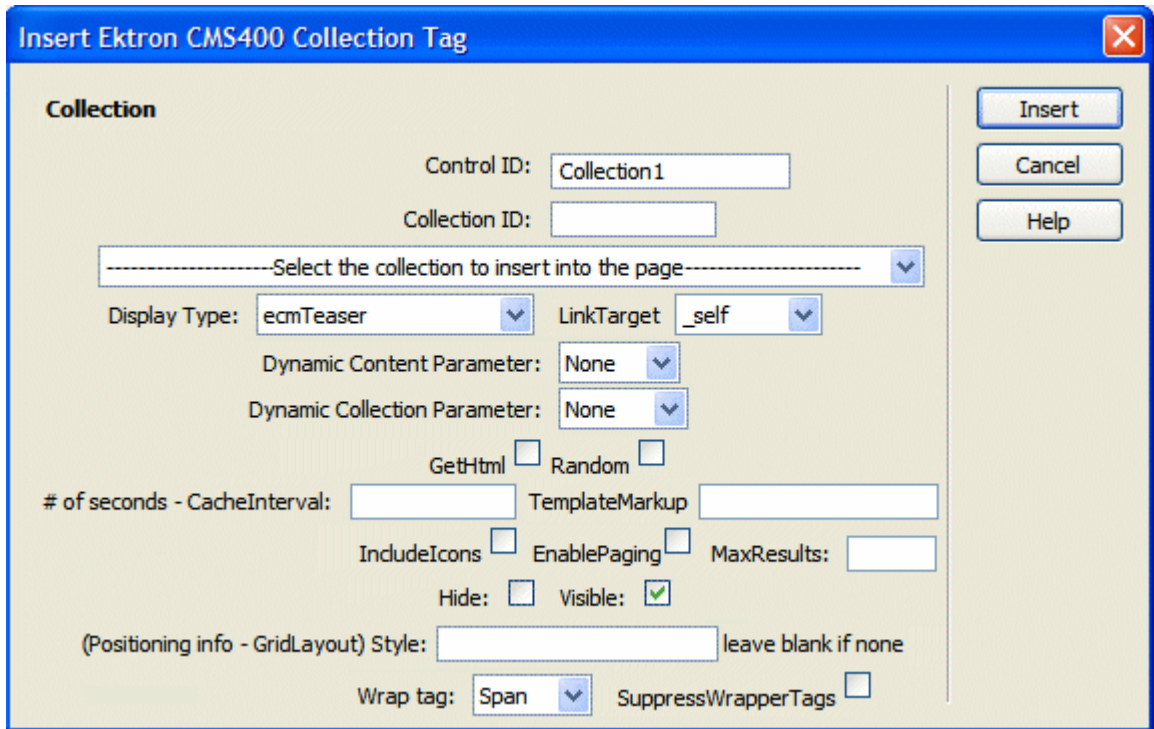
For a description of the Collection function, see ["Collection Server Control" on page 95](#)

This tag appears on the following menu(s):

- Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a collection:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Collection**.
3. The Insert Ektron Collection box is displayed.



**Insert Ektron CMS400 Collection Tag**

**Collection**

Control ID:

Collection ID:

-----Select the collection to insert into the page-----

Display Type:  LinkTarget:

Dynamic Content Parameter:

Dynamic Collection Parameter:

GetHtml ☐ Random ☐

# of seconds - CacheInterval:  TemplateMarkup

IncludeIcons ☐ EnablePaging ☐ MaxResults:

Hide: ☐ Visible: ☒

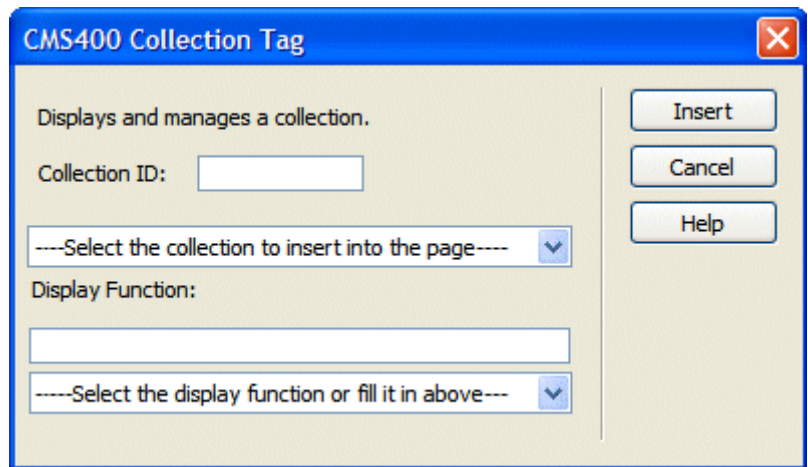
(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:  SuppressWrapperTags ☐

Insert Cancel Help

**IMPORTANT!**

When inserting a collection using ASP, CF, JSP or PHP, a different dialog box appears.



**CMS400 Collection Tag**

Displays and manages a collection.

Collection ID:

-----Select the collection to insert into the page-----

Display Function:

-----Select the display function or fill it in above-----

Insert Cancel Help



## 4. Use the following table to assist you with inserting a Collection.

Field	Description
Control ID <hr/> <b>Used in ASP.NET only.</b> <hr/>	A unique name for the function you are inserting.
Collection ID	If you know the collection ID, enter it into the textbox. If you do not know the Collection ID, you can navigate to it using the Select Collection drop down box.
Select Collection	The drop-down list is filled with a list of all the Collection titles that you have privileges to. When a title is selected, the Collection ID textbox is filled with that Collection's id number. If you insert the Collection ID, you do not need to select a category.
DisplayXslt	<p>Determines how the information on the page is displayed.</p> <p><b>None</b> - databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the collection See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the collection plus the content summary See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>
LinkTarget <hr/> <b>Used in ASP.NET only.</b> <hr/>	<p>Defines the way a link acts when a link is clicked.</p> <p>Choices are:</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned BASE target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "_self" if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "_self" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>

Field	Description
Dynamic Content Parameter <hr/> Used in ASP.NET only.	Checks the QueryString for this value and replaces the collection with a content block when specified. Leave blank to always display the collection.
Dynamic Collection Parameter <hr/> Used in ASP.NET only.	To make this collection dynamic, select <b>coll_id</b> . When you do, the function uses the collection passed as a URL parameter.
GetHtml <hr/> Used in ASP.NET only.	Check this box if you want to retrieve and display content (html body) for all content blocks in the collection. For example, to display content inside a web server control such as a GridView.  <input checked="" type="checkbox"/> <b>Checked</b> = retrieve and display content (html body) for all content blocks in the collection  <input type="checkbox"/> <b>Unchecked</b> = do not retrieve and display content (html body) for all content blocks in the collection
Random <hr/> Used in ASP.NET only.	Check this box if you want to randomly display one collection item. The item changes each time a site visitor views the page.  <input checked="" type="checkbox"/> <b>Checked</b> = randomly display one collection item  <input type="checkbox"/> <b>Unchecked</b> = display collection list
# of seconds - CacheInterval <hr/> Used in ASP.NET only.	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>  <hr/> <b>Important!</b> If the <code>EnablePaging</code> property is set to <code>True</code> , the <code>CacheInterval</code> property is disabled.

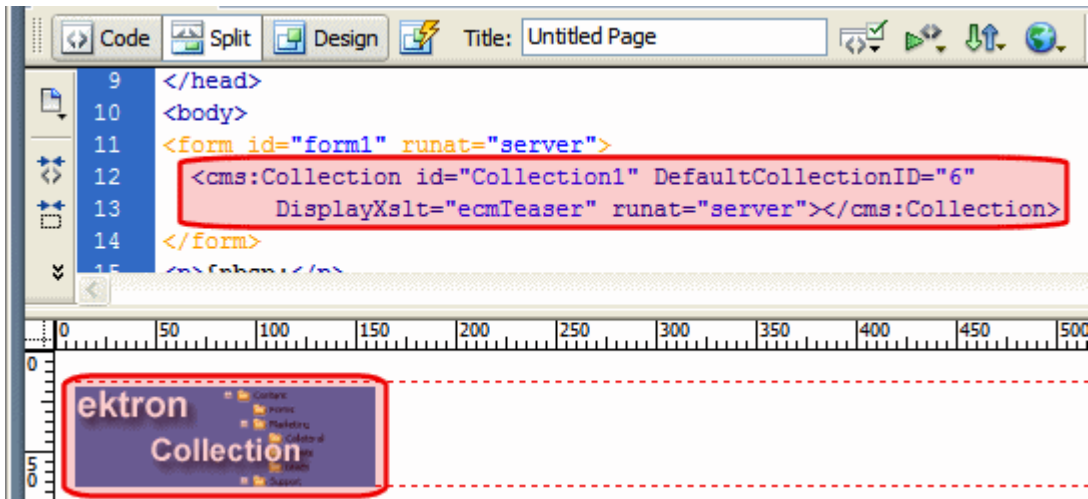
Field	Description
MarkupLanguage <hr/> Used in ASP.NET only.	<p>Identify the template markup file that controls the display of the collection. For example, mycollectionmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \cms400Min\workarea\customfiles\markup\mycollectionmarkup.ekml.</p> <p>See Also: "Ektron Markup Language" on page 754 and "collection.ekml" on page 778</p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>
IncludeIcons <hr/> Used in ASP.NET only.	<p>Choose whether to display icons next to the collection list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Imagelcon] variable is used in an EkML file.</p> <hr/> <div data-bbox="656 829 1188 1221" data-label="Image"> </div>

Field	Description
<p>EnablePaging</p> <hr/> <p>Used in ASP.NET only.</p> <hr/>	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of collection items while controlling the amount of screen space. To accomplish this, the collection display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you set this property box is checked, and the number of collection items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="721 476 1135 706" data-label="Image"> </div> <p>So, for example, if a collection has 9 items and the <code>MaxResults</code> property is set to 4, the screen displays only the first four items. When the site visitor clicks <b>[Next]</b>, he sees items 5, 6, 7 and 8, etc.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = enable paging</p> <p><input type="checkbox"/> <b>Unchecked</b> = disable paging</p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>
<p>MaxResults</p> <hr/> <p>Used in ASP.NET only.</p> <hr/>	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, check the <code>EnablePaging</code> property box.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="721 1392 1135 1621" data-label="Image"> </div>

Field	Description
Hide <hr/> <i>Used in ASP.NET only.</i> <hr/>	Used to hide a collection in design time and run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide collection <input type="checkbox"/> <b>Unchecked</b> = Show collection <hr/> <i>Note: If Hide and Visible are both checked, the property is hidden.</i> <hr/>
Visible <hr/> <i>Used in ASP.NET only.</i> <hr/>	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> <i>Note: If Hide and Visible are both checked, the property is hidden.</i> <hr/>
Style (Positioning Info - GridLayout) <hr/> <i>Used in ASP.NET only.</i> <hr/>	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag <hr/> <i>Used in ASP.NET only.</i> <hr/>	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper <hr/> <i>Used in ASP.NET only.</i> <hr/>	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

- When completed, click **insert**.

A graphic defines where the collection appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:Collection id="Collection1" DefaultCollectionID="6" DisplayXslt="ecmTeaser" runat="server"&gt;&lt;/cms:Collection&gt;</code>
ASP	<code>&lt;% ecmCollection 6,"ecmNavigation" %&gt;</code>
ColdFusion	<code>&lt;cfmodule id="6" displayFunction="ecmNavigation" template="#request.ecm.AppPath#ecmCollection.cfm"&gt;</code>
JSP	<code>&lt;%= ecmCollection(6,"ecmTeaser") %&gt;</code>
PHP	<code>&lt;?php ecmCollection(6,"ecmNavigation"); ?&gt;</code>

## Content Block

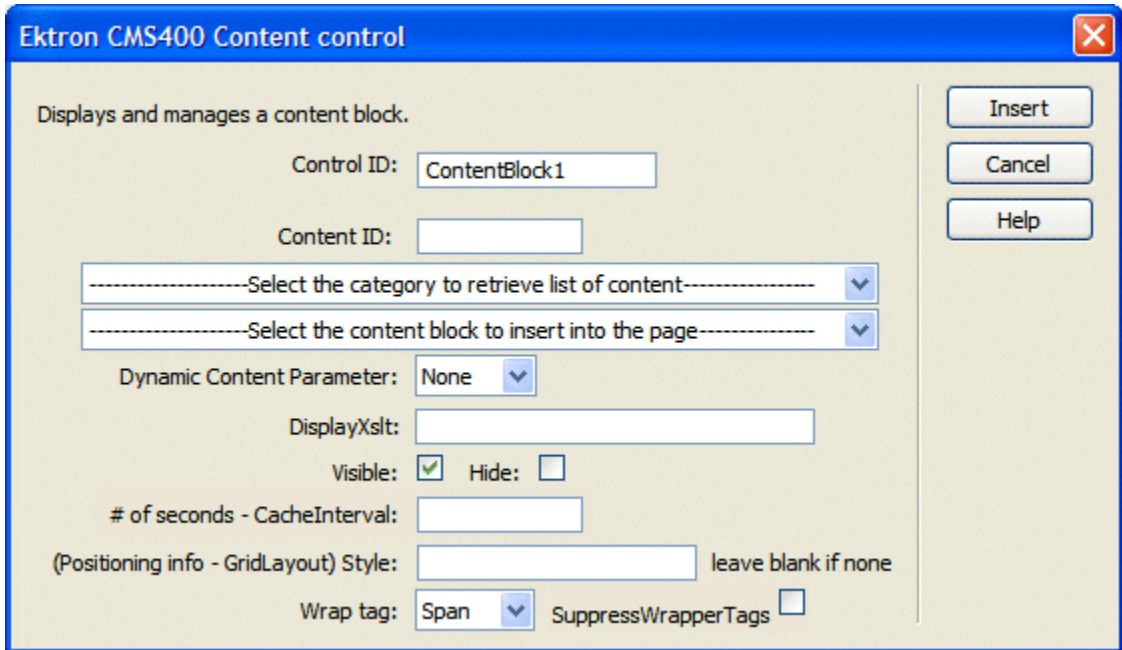
This tag appears on the following menu(s):

- Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

For a description of the Content Block function, see ["ContentBlock Server Control" on page 108](#).

To insert a content block:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Content**.
3. The Insert Content box is displayed



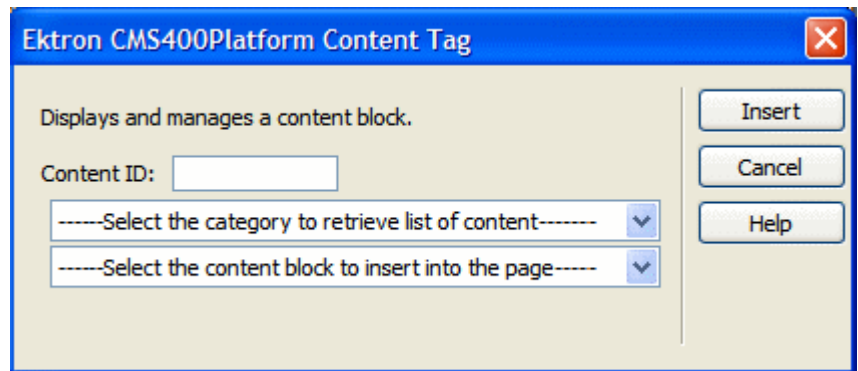
The dialog box is titled "Ektron CMS400 Content control" and has a close button (X) in the top right corner. It contains the following fields and controls:

- Control ID:
- Content ID:
- Select the category to retrieve list of content-----
- Select the content block to insert into the page-----
- Dynamic Content Parameter:
- DisplayXslt:
- Visible: ☒ Hide: ☐
- # of seconds - CacheInterval:
- (Positioning info - GridLayout) Style:  leave blank if none
- Wrap tag:   SuppressWrapperTags ☐

On the right side, there are three buttons: Insert, Cancel, and Help.

**IMPORTANT!**

When inserting a Content Block using ASP, CF, JSP or PHP, a different dialog box appears.



The dialog box is titled "Ektron CMS400Platform Content Tag" and has a close button (X) in the top right corner. It contains the following fields and controls:

- Content ID:
- Select the category to retrieve list of content-----
- Select the content block to insert into the page-----

On the right side, there are three buttons: Insert, Cancel, and Help.

## 4. Use the following table to help you insert a content block.

Field	Description
Control ID <hr/> <i>Used in ASP.NET only.</i> <hr/>	A unique name for the function you are inserting.
Content ID	Specify the ID number of the content block that you wish to use as a content block on the Web page. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content drop down boxes.
Select Category	Choose the folder where the content is located. If you insert the content ID, you do not need to select a category.
Select Content	Select the content block to use from the list. If you insert the content ID, you do not need to select content.
Dynamic Content Parameter <hr/> <i>Used in ASP.NET only.</i> <hr/>	<p>Determines if content is displayed as static or passed dynamically. By making content dynamic, when the content displayed on a Web page contains a link to another piece of content, you can display the second piece of content in the same Web page space.</p> <p><b>None:</b> Displays a static content block.</p> <p><b>id:</b> Content block is passed as a URL parameter.</p> <hr/> <p><b>Important:</b> If you are using an ASP, CF, JSP or PHP page, you must use the Dynamic Content Block Tag to create a dynamic content block. See "Dynamic Content Block" on page 495</p> <hr/>
DisplayXslt <hr/> <i>Used in ASP.NET only.</i> <hr/>	Specify an external XSLT file. See Also: "The OverrideXslt and DisplayXslt Properties" on page 113
Hide <hr/> <i>Used in ASP.NET only.</i> <hr/>	<p>Used to hide a collection in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide collection</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show collection</p> <hr/> <p><b>Note:</b> If Hide and Visible are both checked, the property is hidden.</p> <hr/>

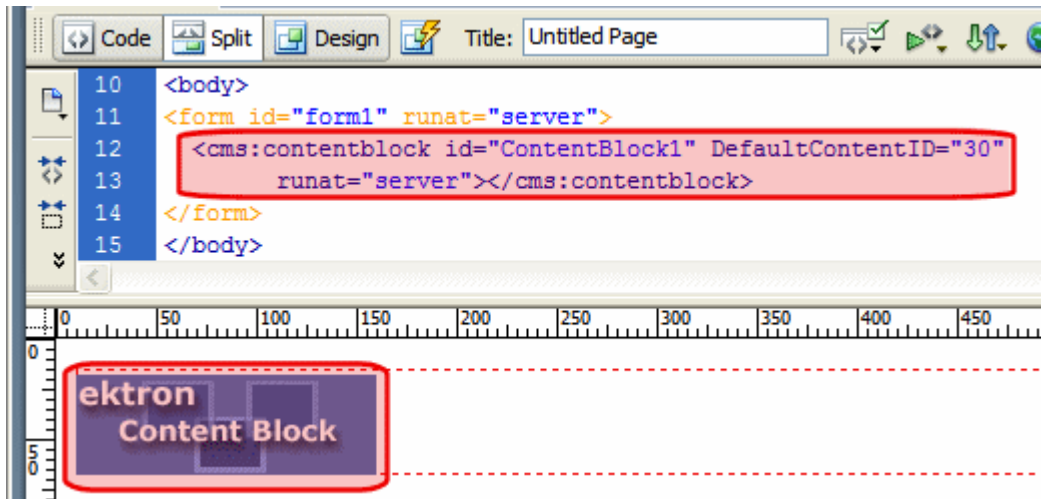


Field	Description
Visible <hr/> Used in ASP.NET only.	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> Note: If Hide and Visible are both checked, the property is hidden.
# of seconds - CacheInterval <hr/> Used in ASP.NET only.	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Style (Positioning Info - GridLayout) <hr/> Used in ASP.NET only.	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag <hr/> Used in ASP.NET only.	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.
SuppressWrapper <hr/> Used in ASP.NET only.	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

**NOTE** You see only the list of Ektron CMS400.NET content blocks if you are working online.

5. Click **insert**.

A graphic indicates the static content block's location on the template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:contentblock id="ContentBlock1" DefaultContentID="12" runat="server"&gt;&lt;/cms:contentblock&gt;</code>
ASP	<code>&lt;% ecmContentBlock(1) %&gt;</code>
ColdFusion	<code>&lt;cfmodule id="1" template="#request.ecm.AppPath#ecmContentBlock.cfm"&gt;</code>
JSP	<code>&lt;%= ecmContentBlock(12) %&gt;</code>
PHP	<code>&lt;?php ecmContentBlock(1); ?&gt;</code>

## Content List

For a description of the Content List function, see ["ContentList Server Control" on page 117](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert the Content List function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Content List**.

3. The Insert Ektron CMS400.NET Content List box appears.

**Ektron CMS400 Content List**

ContentList is used to display a list of content block

Control ID:

ContentIds  Most often leave blank

---

ContentList control can programagically read data out of Content Block vs setting the ContentIds

MetaTag  - of Type Content Selector

DefaultContentID

-----Select the category to retrieve list of content-----

-----Select the content block to insert into the page-----

Dynamic Content Parameter

Random: ☐ GetHtml: ☐

---

Order By:   Direction

DisplayXslt   LinkTarget

# of seconds - CacheInterval:  MaxResults

IncludeIcons ☐ EnablePaging ☐

TemplateMarkup  Visible: ☒ Hide: ☐

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:   SuppressWrapperTags ☐

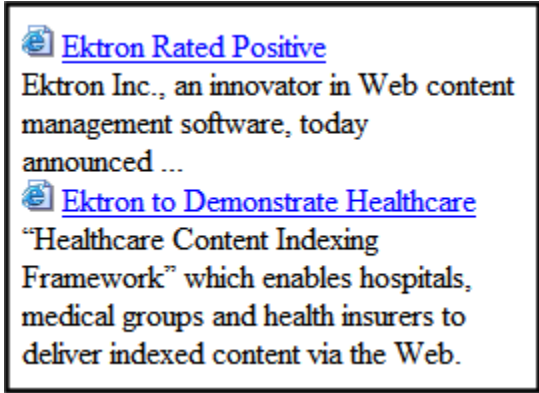
Insert Cancel Help

4. Use the following table to assist you with inserting the Content List function.

Field	Description
Control ID	A unique name for the function you are inserting.
ContentIds	A comma delimited list of content blocks IDs. See Also: <a href="#">"Using the ContentID Property to Display a Content List" on page 126</a>

Field	Description
MetaTag	<p>Specify a Metadata definition whose type is Content Selector. When you do, the associated list of content items will appear where you place the server control.</p> <hr/> <p><b>Warning!:</b> You cannot insert other metadata types.</p> <hr/> <p>This works with the DefaultContentID property.  See Also: <a href="#">"Using the DefaultContentID and MetaTag Properties" on page 122</a>  For more information about using metadata to assign a list of related content to a content item, see the <b>Ektron CMS400.NET</b> Administrator manual section "Managing Content" &gt; "Metadata" &gt; "Types of Metadata" &gt; "Related Content Metadata."</p>
DefaultContentID	<p>Set content id value. Once set, content IDs are generated from the MetaTag value for this content. If you do not know the content's ID, you can navigate to it using the Select Category and Select Content drop down boxes.</p>
DynamicParameter	<p>To make the content dynamic, select <b>id</b>. When you do, this server control uses the content passed as a URL parameter.</p>
Random	<p>Check this box if you want to randomly display one content list item. The item changes each time a site visitor views the page.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = randomly display one content list item</p> <p><input type="checkbox"/> <b>Unchecked</b> = display the content list</p>
GetHtml	<p>Check this box if you want to retrieve and display content (html body) for all content blocks in the content list. For example, to display content inside a web server control such as a GridView.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = retrieve and display content (html body) for all content blocks in the content list</p> <p><input type="checkbox"/> <b>Unchecked</b> = do not retrieve and display content (html body) for all content blocks in the content list</p>

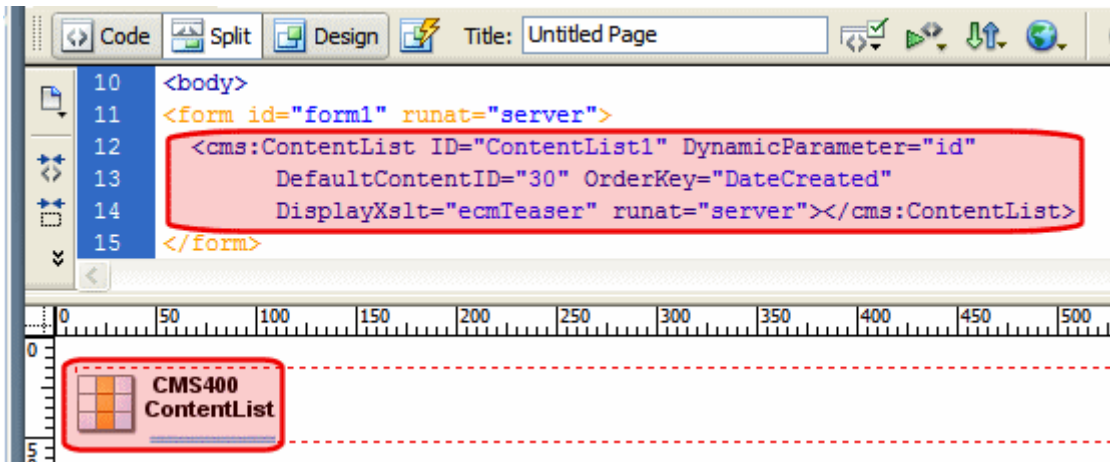
Field	Description
Order By	<p>Sort the list by one of the values below.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - the order of the content's title.</li> <li>• <b>Date Modified</b> - the last date the content was modified.</li> <li>• <b>Date Created</b> - the date the content was created.</li> <li>• <b>Last Author First Name</b> - the last editor's first name.</li> <li>• <b>Last Author Last Name</b> - the last editor's last name.</li> <li>• <b>OrderOfTheIds</b> - preserves the content ID order based on the list in the ContentIds property.</li> </ul>
Direction	Choose whether the list is sorted in Ascending or Descending order.
DisplayXslt	<p>Determines how information on the page is displayed.</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of each content block  See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of each content block plus the content summary  See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <p><b>ecmUnOrderedList</b> - sorts the list in no particular order. Shows the title and content summary</p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "_self" if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "_self" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>

Field	Description
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). <i>See Also:</i> <a href="#">"Caching with Server Controls" on page 42.</a>
Includelcons	<p>Choose whether to display icons next to the content list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Imagelcon] variable is used in an EkML file.</p> <hr/> 
MarkupLanguage	<p>Identify the template markup file that controls the display of the content list. For example, mycontentlistmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \cms400Min\workarea\customfiles\markup\mycontentlistmarkup.ekml.</p> <p><i>See Also:</i> <a href="#">"Ektron Markup Language" on page 754</a> and <a href="#">"contentlist.ekml" on page 781</a></p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>

Field	Description
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Content List icon appears and indicates the Content List's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:ContentList ID="ContentList1" DynamicParameter="id" DefaultContentID="30" OrderKey="DateCreated" DisplayXslt="ecmTeaser" runat="server"&gt;&lt;/cms:ContentList&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Content Rating

For a description of the Content Rating function, see ["ContentRating Server Control" on page 134](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert the Content Rating function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **ContentRating**.



3. The Insert Ektron CMS400.NET Content Rating box appears.

4. Use the following table to assist you with inserting the Content Rating function.

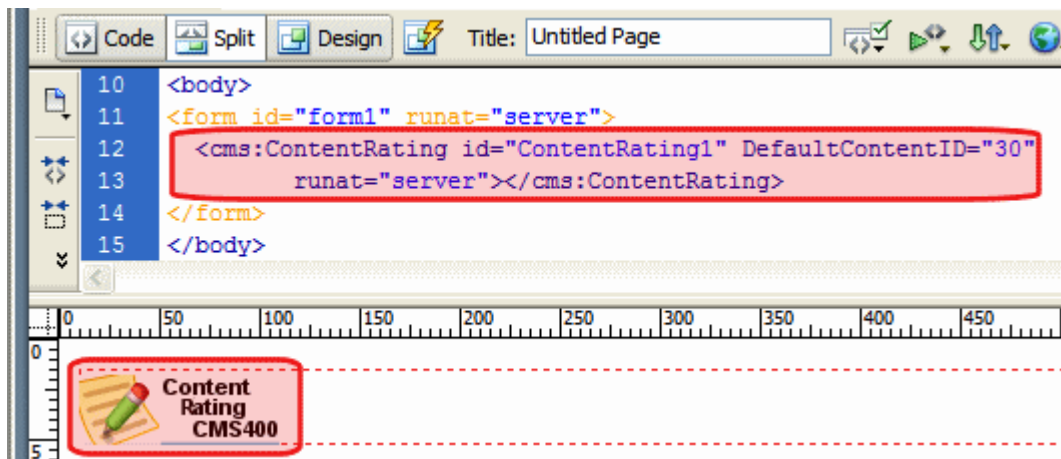
Field	Description
Control ID	A unique name for the function you are inserting.
Content ID	Specify the ID number of a content item being tracked by this function. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content drop down boxes.
Dynamic Content Parameter	To make this content rating control dynamic, select <b>id</b> . When you do, this server control is attached to the content block passed as a URL parameter.
CompleteGraph Display	Specify a value to indicate how the ratings graph appears if a site visitor has already rated content. <ul style="list-style-type: none"> <li>• <b>horizontal</b> - graph appears to the right of the AlreadyRatedMessage</li> <li>• <b>vertical</b> - graph appears below the AlreadyRatedMessage</li> <li>• <b>none</b> - graph does not appear</li> </ul>

Field	Description
GraphBarColor	The color of the bars in the graph.
InitialGraphDisplay	<p>Specify a value to indicate how the ratings graph appears if a site visitor has not yet rated content.</p> <ul style="list-style-type: none"> <li>• <b>horizontal</b> - graph appears to the right of the content rating scale</li> <li>• <b>vertical</b> - graph appears below the content rating scale</li> <li>• <b>none</b> - graph does not appear</li> </ul>
Padding	<p>Enter the number of pixels used to create space between the rating scale and the graph.</p> <hr/> <p><a href="#">Note: If you do not display the graph, this setting has no effect.</a></p> <hr/>
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>

Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

An Content Rating icon appears and indicates the Content Rating's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:ContentRating id="ContentRating1" DefaultContentID="30" runat="server"&gt;&lt;/cms:ContentRating&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.

Platform	Code View
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Content XSLT Tag...

The insert content XSLT command allows you to insert an XML content block, and specify an XSLT (other than the default) to be applied to it.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > XML Function...**

To insert a content XSLT tag, perform the following steps:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET XML functions, click on **Insert Content XSLT Tag...**
3. The Insert Ektron CMS400.NET Content and Apply an XSLT Tag box is displayed.

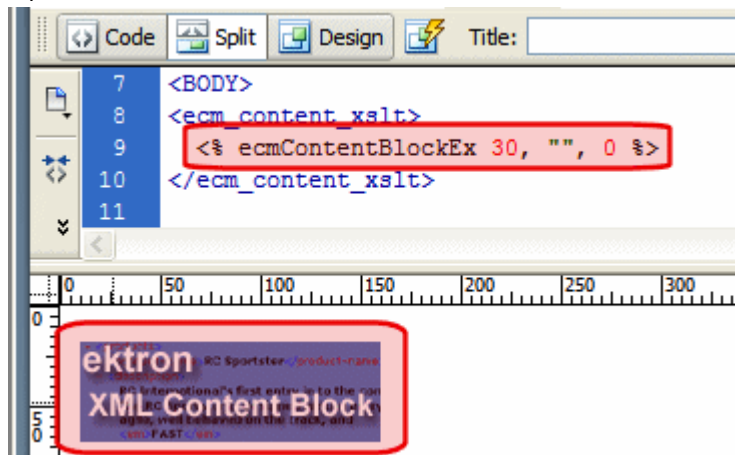
4. Use the following table to assist you with this form.

Field	Description
Content ID	Either type in, or select from the drop down lists, the ID number of the XML content to insert.
Select XSLT	From the drop down list, select the XSLT you would like to apply to the XML content block.

Field	Description
Enter XSLT	If "Enter XSLT" is selected, you can insert the path and file name of the XSLT to apply to the content.

5. Click **insert**.

A graphic showing you where the XML content block will be placed on the template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
.NET	This function is not available as an Ektron Dreamweaver Extension for .NET.
ASP	<code>&lt;% ecmContentBlockEx 30, "", 1 %&gt;</code>
ColdFusion	<code>&lt;cfmodule id="5" xsltpath="" xslttype="1" template="#request.ecm.AppPath#ecmContentBlockEx.cfm"&gt;</code>
JSP	<code>&lt;%= ecmContentBlockEx(24, "", 0 ) %&gt;</code>
PHP	<code>&lt;?php ecmContentBlockEx (5, "", 1); ?&gt;</code>

## DHTML Menu

For a description of the DHTML Menu function, see ["DhtmlMenu Server Control" on page 254](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS MENU Control...**

To insert a DHTML Menu:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **DHTML Menu**  
The DHTML Menu Control box appears.

**Ektron CMS400 DhtmlMenu Control**

DhtmlMenu

Control ID:

DefaultMenuID:

-----Select the from a list of menus-----

Align : HorizontalDisplay: ☐

CssClass:

MenuHoverOverColor  : SubMenuBackColor:

SubMenuHoverOverColor  : BackColor

ForeColor  : Font-Bold ☐ Font\_Italic ☐

# of seconds - CacheInterval::

FontNames  : Font-Size

SubMenuItemHeight  : SubMenuItemWidth

SubMenuTopOffset  : MenuStyle:

Stylesheet:

Dynamic menu Parameter:

Wrap tag SuppressWrapperTags ☐ Visible: ☒ Hide ☐

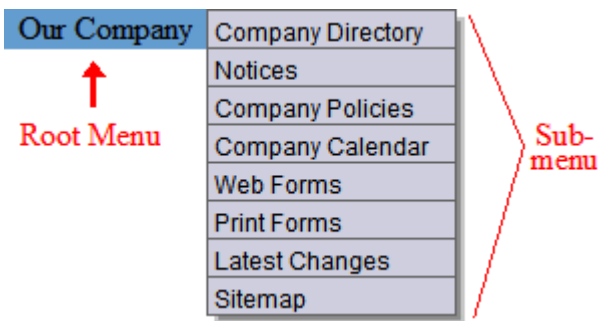
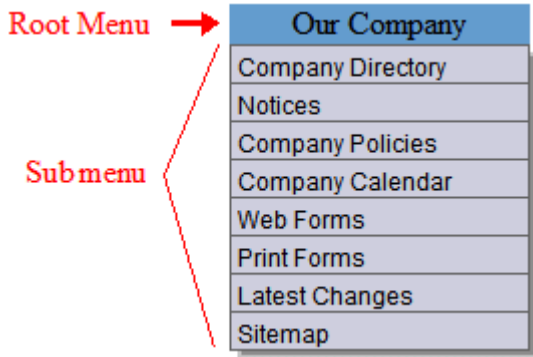
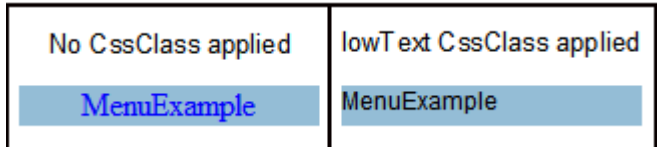
(Positioning info - GridLayout) Style:  leave blank if none

Insert Cancel Help

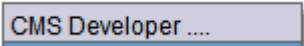

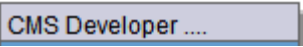

3. Use the following table to assist you with inserting a DHTML Menu.

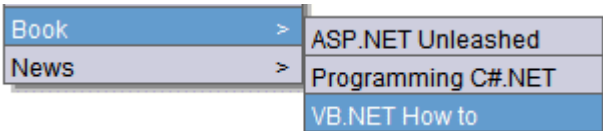
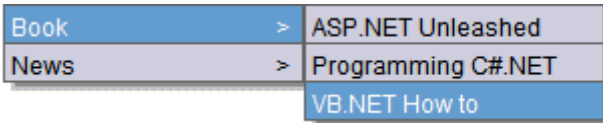
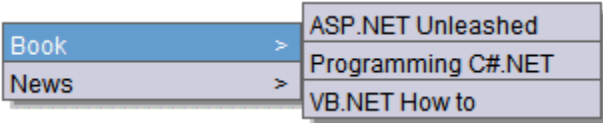

Field	Description
Control ID	A unique name for the function you are inserting.

Field	Description
DefaultMenuID	The numeric id of the menu that you want to insert. If you do not know the DefaultMenuID path, you can navigate to it using the Select Menu drop down box.
Select Menu	Select a menu to be used. When a menu is selected, the DefaultMenuID textbox is filled with that menu's id number. If you insert the DefaultMenuID, you do not need to select a menu.
Align	<p>Specifies the horizontal alignment of the root menu text.</p> <p>The default is <b>Center</b>.</p> <p><b>NotSet</b> - Use the default setting.</p> <p><b>Left</b> - Aligns text to the left.</p> <p><b>Center</b> - Aligns text to the center.</p> <p><b>Right</b> - Aligns text to the right.</p> <p><b>Justify</b> - Aligns text justified to both margins.</p>

Field	Description
HorizontalDisplay	<p>Displays submenu to the right of the root menu. This only affects the first submenu of the root menu. All subsequent submenus appear to the right of each submenu.</p> <p>The default is Unchecked</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Submenu appears to the right of the root menu</p>  <p><input type="checkbox"/> <b>Unchecked</b> = Submenu appears below the root menu</p> 
CssClass	<p>Apply a .css class to the root menu. Enter the class name from the .css defined in the Style sheet property.</p> <p>Since pop_style.css is the default style sheet, you can use its classes without defining it in the Stylesheet property.</p> <p>For example, if you enter lowtext in the property, the root menu changes as shown below.</p> 



Field	Description
MenuHoverOverColor	Sets the background color for the root menu when hovering over it.
SubMenuBackColor	Sets the background color for the submenu.
SubMenuHoverOverColor	Sets the background color for the submenu when hovering over it.
BackColor	Sets the background color for the root menu. This property appears in the <b>Appearance</b> category
ForeColor	Sets the color of the font for the root menu. This property appears in the <b>Appearance</b> category.
Font-Bold	Checking this box allows the font to appear <b>bold</b> .
Font_Italic	Checking this box allows the font to appear <i>italics</i> .
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: " <a href="#">Caching with Server Controls</a> " on page 42.
FontNames	Enter the name of a font to use.
Font-Size	The size of the font.
SubMenuItemHeight	<p>Sets the height of each submenu item in pixels. The Default is 20</p> <p>Example set to 20: </p> <p>Example set to 30: </p>
SubMenuItemWidth	<p>Sets the width of each submenu item in pixels. The Default is 150</p> <p>Example set to 150: </p> <p>Example set to 130: </p>

Field	Description
SubMenuTopOffset	<p>Sets the location of the top of the submenu. This number is in pixels. The default is 3</p> <p>Example set to 5:</p>  <p>Example set to 0 (zero):</p>  <p>Example set to -10:</p> 
MenuStyle	<p>A specified inline style that defines the root menu.</p> <p>For example, if you enter the following in the MenuStyle property box:</p> <pre>font: bold 18px Arial;color:royalblue; background-color:slateblue</pre> 
StyleSheet	<p>Specify the location and style sheet for the submenu. For example:</p> <pre>\Workarea\csslib\mytest.css.</pre> <p>Leave blank to use the default.</p> <p>pop_style.css is the default style sheet. It is located in \webroot\CMS400Min\Workarea\csslib. For more information on style sheets, see <a href="#">"Using a Style Sheet" on page 260</a></p> <p>The Stylesheet property can be used in conjunction with the CssClass property to define the root menu.</p>
Dynamic Menu Parameter	<p>Checks the QueryString for the menu value and replaces the menu with content when ID is specified. Leave blank to always display the menu.</p> <p><b>None</b> - Use Default</p> <p><b>menu_ID</b> - The control uses the menu passed as a URL parameter.</p>

Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>

4. When completed, click **insert**.

A graphic defines where the DHTML Menu appears on your template.

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<pre>&lt;cms:DhtmlMenu id="DhtmlMenu1" DefaultMenuID="16" ForeColor="" runat="server"&gt;&lt;/cms:DhtmlMenu&gt;</pre>

Platform	Code View
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Directory-Taxonomy

For a description of the Directory-Taxonomy function, see "[DhtmlMenu Server Control](#)" on page 254.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a Directory-Taxonomy:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Directory-Taxonomy**  
The Directory-Taxonomy Control box appears.

**Ektron CMS400 Directory - Taxonomy**

**Directory - Taxonomy**

Control ID:

TaxonomyId  ItemsControlId

TaxonomyCols  TaxonomyDepth

DynamicParameter  LinkTarget

TaxonomyStyle  MaxResults

EnableAjax ☒ EnablePaging ☒

TaxonomySearch ☒ DisplayXslt

NaviDisplayControlId

# of seconds - CacheInterval:  TemplateMarkup

Hide: ☐ Visible: ☒

(Positioning info - GridLayout) Style:   
leave blank if none

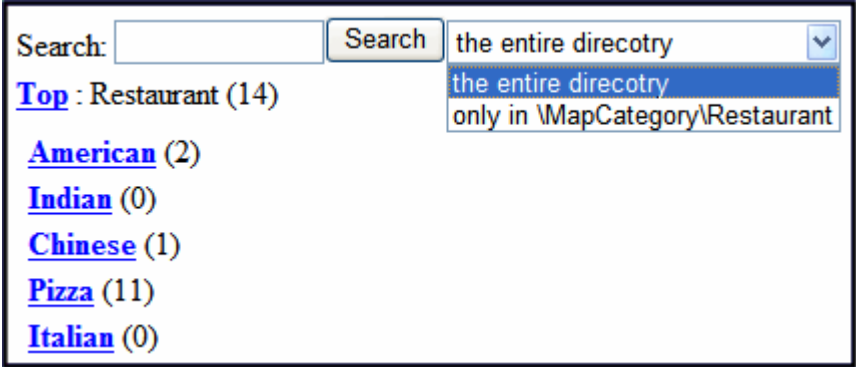
Wrap tag:  SuppressWrapperTags ☐

3. Use the following table to assist you with inserting a Directory-Taxonomy.

Field	Description
Control ID	A unique name for the function you are inserting.
TaxonomyId	Enter the ID number of the taxonomy or category to appear in this server control. If you don't know the number, click the button and navigate to the taxonomy or category. When you select one, it appears in the center of the Visual Studio window.
ItemsControlId	Specify the id of the <code>&lt;span&gt;</code> or <code>&lt;div&gt;</code> tag you will use to format the display of taxonomy items.

Field	Description
TaxonomyCols	Enter the number of columns in which this taxonomy/category will appear on the page.
TaxonomyDepth	<p>Enter the number of content items to appear below each taxonomy/category. Following this number of items, three periods appear to indicate more items are available.</p> <p>important for performance as well as display -- if you enter 0, all content under taxonomy copied from server to client -- could be slow</p> <p>if you enter 1, just one node under the top level is copied to client -- testing on this is inconclusive</p>
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.
Link Target	<p>Defines the way a link acts when it is clicked. The choices are:</p> <p><b>_blank</b> - loads the link in a new blank window. This window is not named.</p> <p><b>_self</b> - loads the link in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - loads the link in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p><b>_top</b> - loads the link in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>
MaxResults	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, check the <code>EnablePaging</code> property box.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="685 1308 1099 1536" data-label="Image"> <p>The screenshot shows a rectangular box containing a list of four names, each on a new line and underlined in blue: Bill Lumbergh, Bob Porter, Bob Slydell, and Milton Waddams. Below the names, there are four pagination links: [First], [Previous], [Next], and [Last], all in blue text.</p> </div>

Field	Description
EnableAjax	<p>Check this box to enable Ajax searches. When enabled, the <code>MaxResults</code> property determines the maximum number of results per page.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Enable Ajax Search</p> <p><input type="checkbox"/> <b>Unchecked</b> = Original HTML Search</p>
EnablePaging	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of taxonomy items while controlling the amount of screen space. To accomplish this, the taxonomy display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you check this property box, and the number of taxonomy items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="686 682 1099 910" data-label="Image"> </div> <p>So, for example, if a taxonomy has 9 items and the <code>MaxResults</code> property is set to 3, the screen displays only the first three items. When the site visitor clicks <b>[Next]</b>, he sees items 4, 5 and 6, etc.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = enables paging</p> <p><input type="checkbox"/> <b>Unchecked</b> = disables paging</p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to True, the <code>CacheInterval</code> property is disabled.</p> <hr/>

Field	Description
TaxonomySearch	<p>Check this box if you want a search box (shown below) to appear above this taxonomy display. A site visitor can use the search to find content within a taxonomy/category that includes terms of interest.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = a search box appears above the taxonomy display</p> <p><input type="checkbox"/> <b>Unchecked</b> = suppress the search box</p> 
DisplayXslt	<p>Determines how the information on the page is displayed. Options are</p> <p><b>None</b> - databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder  See Also: "<a href="#">Example of ecmNavigation Display</a>" on page 101</p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary  See Also: "<a href="#">Example of ecmTeaser Display</a>" on page 102</p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p><b>Note:</b> If an EkML file is inserted into the MarkupLanguage property, this value is ignored.</p> <hr/>
NaviDisplayControllId	<p>Specify the id of the &lt;span&gt; or &lt;div&gt; tag to format the display of this taxonomy's navigation. Here is an example.</p> <p><b>Top : <u>Restaurant</u></b> : Italian (0)</p>

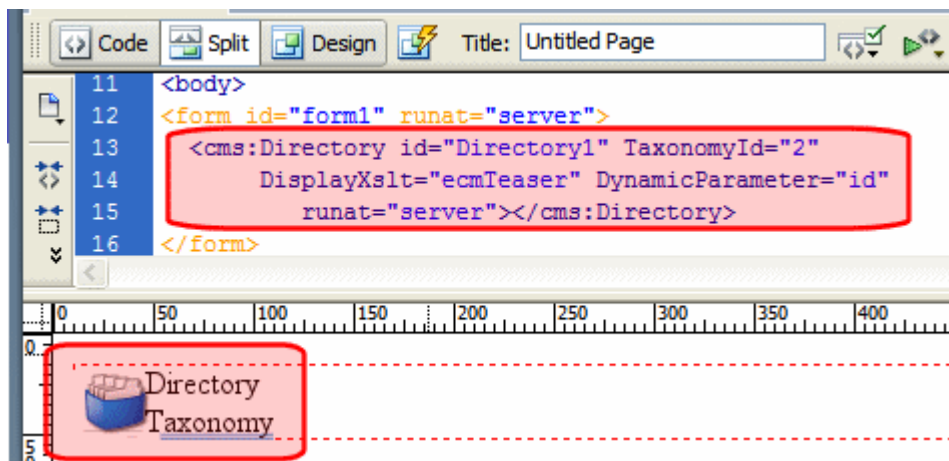


Field	Description
# of seconds - CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a></p> <hr/> <p><b>Important!</b> If the EnablePaging property is set to True, the CacheInterval property is disabled.</p> <hr/>
MarkupLanguage	<p>Identify the template markup file that controls the display of the taxonomy. For example, mytaxonomymarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \cms400Min\workarea\customfiles\markup\mytaxonomymarkup.ekml.</p> <p>See Also: <a href="#">"Ektron Markup Language" on page 754</a> and <a href="#">"taxonomy.ekml" on page 794</a></p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>

Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

4. When completed, click **insert**.

A graphic defines where the Directory-Taxonomy appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Directory id="Directory1" TaxonomyId="2" DisplayXslt="ecmTeaser" DynamicParameter="id" runat="server"&gt;&lt;/cms:Directory&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.

Platform	Code View
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Discussion Forum

For a description of the Discussion Forum function, see "[Discussion Board Server Controls](#)" on page 147.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Forums**

To insert the Discussion Forum function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Discussion Forum**.
3. The Insert Ektron CMS400.NET Discussion Forum box appears.

4. Use the following table to assist you with inserting the Blog function.

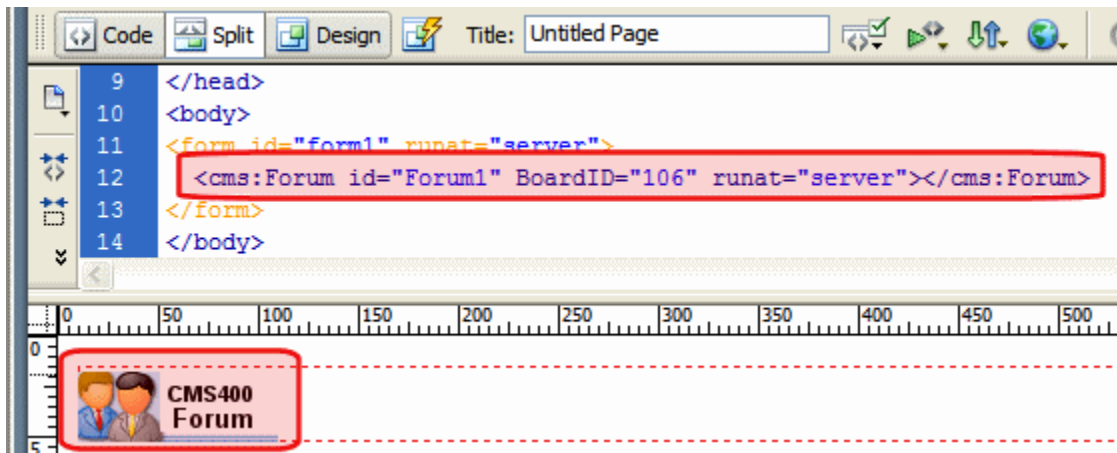
Field	Description
Control ID	A unique name for the function you are inserting.
Board ID	The ID of the Discussion Board in CMS400.NET. If you do not know the Board ID, you can choose it from the Select Folder drop down box.

Field	Description
JavascriptEditorToolbar	Decide which buttons are available in the Javascript Editor. See Also: "Adding and Removing Toolbar Buttons From the Javascript Editor" on page 151
JavascriptEditorWidth	Set the width in pixels for the editor. The default is <b>625</b> .
JavascriptEditorHeight	Set the height in pixels for the editor. The default is <b>400</b> .
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42.
Hide	<p>Used to hide a function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p>Note: If Hide and Visible are both checked, the property is hidden.</p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p>Note: If Hide and Visible are both checked, the property is hidden.</p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.</p>

Field	Description
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Discussion Forum icon appears and indicates the Discussion Forum's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Forum id="Forum1" BoardID="106" runat="server"&gt;&lt;/cms:Forum&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Display By Meta Value

For a description of the Display By Meta Value function, see ["MetaDataList Server Control" on page 266](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert Display By Meta Value:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Display By Meta Value**  
The Display By Meta Value box appears.

3. Use the following table to assist you with inserting Display By Meta Value.

Field	Description
Control ID	A unique name for the function you are inserting.

Field	Description
FolderID	The numeric id of the folder that you want to use. If you do not know the folder ID, you can navigate to it using the Select folder drop down box.
Select Folder	Select a folder path to be used. When a folder is selected, the folderID textbox is filled with that folder's id number. If you insert the Folder ID, you do not need to select a folder.
Includelcons	<p>Choose whether to display icons next to the collection list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Imagelcon] variable is used in an EkML file.</p> <hr/>
Recursive	Determines whether the Display By Meta Value control uses this folder's child folders.

Field	Description
EnablePaging	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of taxonomy items while controlling the amount of screen space. To accomplish this, the taxonomy display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you check this property box, and the number of taxonomy items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p> <div data-bbox="672 484 1089 711" data-label="Image"> </div> <p>So, for example, if a taxonomy has 9 items and the <code>MaxResults</code> property is set to 3, the screen displays only the first three items. When the site visitor clicks <b>[Next]</b>, he sees items 4, 5 and 6, etc.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = enables paging</p> <p><input type="checkbox"/> <b>Unchecked</b> = disables paging</p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to <code>True</code>, the <code>CacheInterval</code> property is disabled.</p> <hr/>
Order By	<ul style="list-style-type: none"> <li>Specify how you want to order the results that are generated. Also specify if you want the results in ascending or descending order.</li> <li><b>Title</b> - The title of the content block</li> <li><b>Date Created</b> - The date the content block was created</li> <li><b>Date Modified</b> - The date the content block was last modified</li> <li><b>Last Author Last Name</b> - The last author's last name</li> <li><b>Last Author First Name</b> - The last author's first name</li> <li><b>ID</b> - The content blocks ID</li> <li><b>Ascending</b> - Alphabetic or numerical order</li> <li><b>Descending</b> - Reverse alphabetic or numerical order</li> </ul>



Field	Description
MaxResults	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, check the <code>EnablePaging</code> property box.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="668 558 1078 784"> <p>A screenshot of a list of names: <a href="#">Bill Lumbergh</a>, <a href="#">Bob Porter</a>, <a href="#">Bob Slydell</a>, and <a href="#">Milton Waddams</a>. Below the list are four navigation links: <a href="#">[First]</a>, <a href="#">[Previous]</a>, <a href="#">[Next]</a>, and <a href="#">[Last]</a>. The links are enclosed in a black rectangular border.</p> </div>
ExactPhrase	<p>Determines whether the <code>KeywordValue</code> needs to match the phrase exactly.</p> <p>For example, if "site" is the <code>KeywordValue</code> and the title of a content block is "Welcome to the site" and the <code>ExactPhrase</code> box is checked, you would not see the content block in the metadata list. This is because "site" does not equal "Welcome to the site".</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Match the exact phrase</p> <p><input type="checkbox"/> <b>Unchecked</b> = Doesn't need to match exact phrase</p>
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p>Choices are:</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned BASE target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "_self" if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "_self" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>

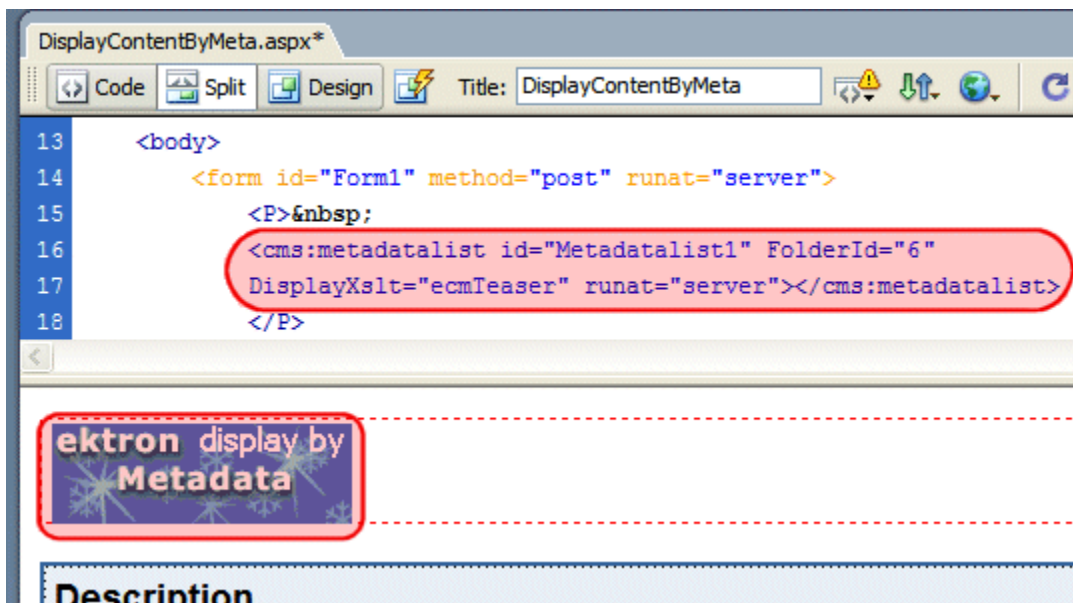
Field	Description
KeyWordName	<p>The KeyWordName is the container where the KeyWordValues are located. Some examples of a KeyWordName might be "Keywords or Title."</p> <p>For information on creating KeyWordNames for Metadata, see the <b>Ektron CMS400.NET</b> Administrators Manual &gt; "Managing Content" &gt; "Metadata" &gt; "Managing Metadata Definitions" &gt; "Adding a Metadata Definitions"</p>
KeyWordValue	<p>The value that is associated with the KeyWordName. Some examples of a KeyWordValue might be "Home; page; or company"</p>
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a></p> <hr/> <p><b>Important!</b> If the EnablePaging property is set to True, the CacheInterval property is disabled.</p> <hr/>
DisplayXslt	<p>Determines how the information on the page is displayed.</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <hr/> <p><b>Important!</b> If you enter a valid EkML file at the MarkupLanguage property, the Displayxslt property value is ignored.</p> <hr/>
MarkupLanguage	<p>Identify the template markup file that controls the display of the metadata list. For example, mymetadatalistmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example,  \cms400Min\workarea\customfiles\markup\mymetadatalist.ekml.</p> <p>See Also: <a href="#">"Ektron Markup Language" on page 754</a> and <a href="#">"metadatalist.ekml" on page 791</a></p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>

Field	Description
Dynamic Content Parameter	<p>Checks the QueryString for the content block ID value and replaces the Display By Meta Value Control with content when an ID is specified. Leave blank to always display the Display By Meta Value Control.</p> <p><b>None</b> - Use Default</p> <p><b>ID</b> - The control uses the content block passed as a URL parameter.</p>
Content Type	<p>Defines what type of list summary appears.</p> <p><b>Content</b> - Displays only content blocks in the folder.</p> <p><b>All Types</b> - Displays all content in the folder.</p> <p><b>Forms</b> - Displays only forms in the folder.</p> <p><b>Archive_Content</b> - Displays archived content from the folder.</p> <p><b>Archive_Forms</b> - Displays archived forms from the folder.</p>
GetHtml	<p>Check this box if you want to retrieve and display content (html body) for all content blocks in the collection. For example, display content inside a web server control such as a GridView.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = retrieve and display content (html body) for all content blocks in the metadata list</p> <p><input type="checkbox"/> <b>Unchecked</b> = do not retrieve and display content (html body) for all content blocks in the metadata list</p>
Hide	<p>Used to hide the function in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p>Note: If Hide and Visible are both checked, the property is hidden.</p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p>Note: If Hide and Visible are both checked, the property is hidden.</p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page. For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>

Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

4. When completed, click **insert**.

A graphic defines where the Display By Meta Value appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<pre>&lt;cms:metadatalist id="Metadatalist1" FolderId="6" DisplayXslt="ecmTeaser" runat="server"&gt;&lt;/cms:metadatalist&gt;</pre>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.

Platform	Code View
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Dynamic Content Block

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

For a description of a dynamic content block, see ["Dynamic Content Block" on page 612](#)

To insert a dynamic content block tag:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Dynamic Tag...**
3. The Insert Ektron Dynamic Tag box is displayed

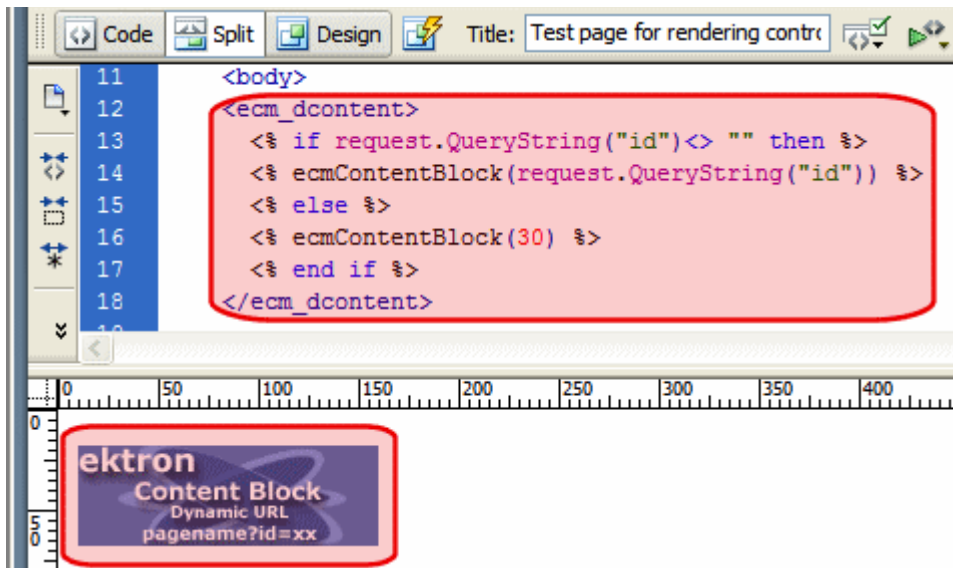
4. Use the following table to assist you with inserting a dynamic content block function.

Field	Description
Insert Default Content	If you want a static content block to be displayed if no ID is passed through the URL, place a check in this box.
Default Content ID	Specify the ID number of the content block that you wish to use as a default content block on the Web page.

Field	Description
Select Content	If you are working online, you may select the content block to use as a default content block from the list.

- When completed, click **Insert**.

A graphic to define where the dynamic content block has been placed on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
.NET	For ASP.NET pages, use the CMS400.NET (Server Control) Content Block tag and set the dynamic parameter to id. See <a href="#">"Content Block" on page 456</a>
ASP	<pre> &lt;% if request.QueryString("id") &lt;&gt; "" then ecmContentBlock(request.QueryString("id")) else ecmContentBlock(1) end if %&gt; </pre>
ColdFusion	<pre> &lt;cfif isdefined("url.id")&gt;     &lt;cfmodule id="#url.id#" template="#request.ecm.AppPath#ecmContentBlock.cfm"&gt;     &lt;cfelse&gt;     &lt;cfmodule id="1" template="#request.ecm.AppPath#ecmContentBlock.cfm"&gt;     &lt;/cfif&gt; </pre>

Platform	Code View
JSP	Inserting a Dynamic Content Block using Ektron's Dreamweaver Extensions is not supported in JSP.
PHP	<code>&lt;?php \$id =GetURL_Id(1); ecmContentBlock(\$id); ?&gt;</code>

## Folder Bread Crumb

For a description of the Folder Bread Crumb function, see ["FolderBreadCrumb Server Control" on page 82](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS MENU Controls...**

To insert a Folder Bread Crumb Trail:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Folder BreadCrumb**.
3. The Folder Bread Crumb dialog box appears.

4. Use the following table to assist you with inserting a Folder Bread Crumb.

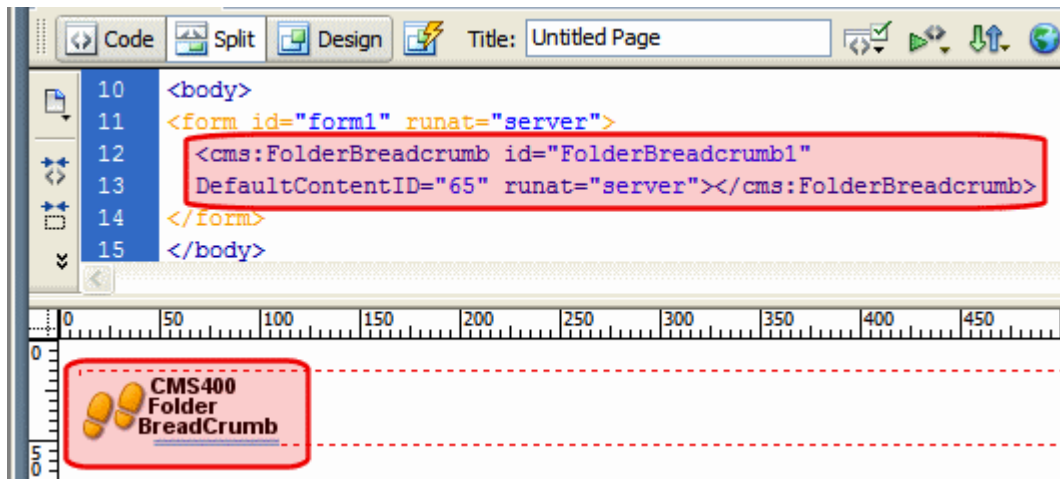
Field	Description
Control ID	A unique name for the function you are inserting.
FolderID	The numeric id of the folder that you want to use. If you do not know the folder ID, you can navigate to it using the Select folder drop down box.
Content ID	Gets or Sets the ContentID. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content drop down boxes.
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.
DisplayStyle	Indicate how to display the folder bread crumb: horizontally or vertically. The default is Horizontal.
Hide	<p>Used to hide a FolderBreadcrumb in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide FolderBreadcrumb</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show FolderBreadcrumb</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>



Field	Description
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. When completed, click **Insert**.

A graphic defines where the Folder Bread Crumb appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<pre>&lt;cms:FolderBreadcrumb id="FolderBreadcrumb1" DefaultContentID="65" runat="server"&gt;&lt;/cms:FolderBreadcrumb&gt;</pre>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Form

For a description of the Form function, see ["FormBlock Server Control" on page 159](#)

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a form:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click **Form**
3. The Insert Form box is displayed.

**IMPORTANT!**

When inserting a form tag using ASP, CF, JSP or PHP, a different dialog box appears.

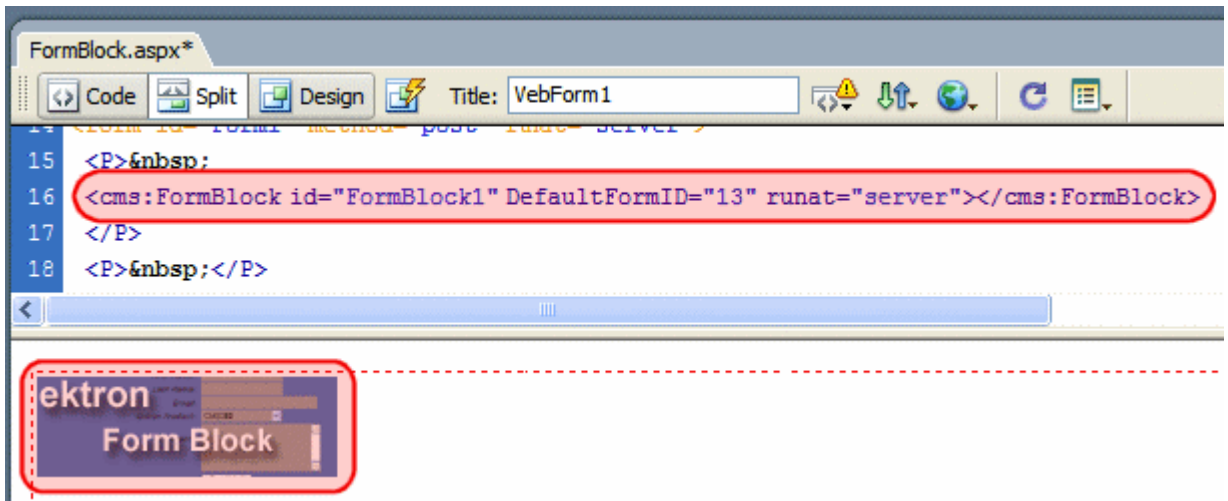
## 4. Use the following table to assist you with inserting a form.

Field	Description
Control ID <u>Used in ASP.NET only.</u>	A unique name for the function you are inserting.
Form ID	Type in the form ID that you want to insert. If you do not know the Form ID, you can navigate to it using the Select Form drop down box.
Select Form	The drop-down box is populated with a list of all the form titles. When a title is selected, the form ID textbox is filled with that form's id number. If you insert the Form ID, you do not need to select a form.
Dynamic Form Parameter <u>Used in ASP.NET only.</u>	To make this form block dynamic, select <b>ekfrm</b> . When you do, this control uses the form block passed as a URL parameter.
# of seconds - CacheInterval <u>Used in ASP.NET only.</u>	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See A/so: <a href="#">"Caching with Server Controls" on page 42</a> .
Add Validation <u>Used in ASP.NET only.</u>	When <b>checked</b> , adds Validation to your main runat=server form. <input checked="" type="checkbox"/> <b>Checked</b> = add validation <input type="checkbox"/> <b>Unchecked</b> = Do not add validation
Include Form Tags <u>Used in ASP.NET only.</u>	Determines if tags are generated automatically or manually. See A/so: <a href="#">"Automatic versus Manual Generation of Form Tags" on page 161</a> <input checked="" type="checkbox"/> <b>Checked</b> = automatically generate form tags <input type="checkbox"/> <b>Unchecked</b> = generate form tags manually
Hide <u>Used in ASP.NET only.</u>	Used to hide a form in design time and run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide form <input type="checkbox"/> <b>Unchecked</b> = Show form  <u>Note: If Hide and Visible are both checked, the property is hidden.</u>

Field	Description
Visible <hr/> <i>Used in ASP.NET only.</i>	Used to show or hide the function at run time.  <input checked="" type="checkbox"/> <b>Checked</b> = Show function  <input type="checkbox"/> <b>Unchecked</b> = Hide function  <hr/> <i>Note: If Hide and Visible are both checked, the property is hidden.</i>
Style (Positioning Info - GridLayout) <hr/> <i>Used in ASP.NET only.</i>	Set position information of a control as absolute, when using a GridLayout Web page.  For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag <hr/> <i>Used in ASP.NET only.</i>	Allows a developer to specify a server control's tag.  The default is <b>Span</b> .  <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element.  <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> .  <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags  <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

- When completed, click **Insert**.

A graphic defines where the form appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:FormBlock id="FormBlock1" DefaultFormID="13" runat="server"&gt;&lt;/cms:FormBlock&gt;</code>
ASP	<code>&lt;% ecmFormBlock (1) %&gt;</code>
ColdFusion	<code>&lt;cfmodule id="1" template="#request.ecm.AppPath#ecmFormBlock.cfm"&gt;</code>
JSP	<code>&lt;%= ecmFormBlock(1) %&gt;</code>
PHP	<code>&lt;?php ecmFormBlock (1); ?&gt;</code>

## ImageControl

For a description of the ImageControl function, see ["AssetControl Server Control" on page 51](#).

This tag appears on the following menu(s):

- Commands > Ektron CMS400.NET (Server Controls) > Community Controls...**

To insert the ImageControl function:

- Move the blinking cursor to the area that you want to place it.
- From the list of Ektron CMS400.NET functions, click **ImageControl**.

3. The Insert Ektron CMS400.NET ImageControl box appears.

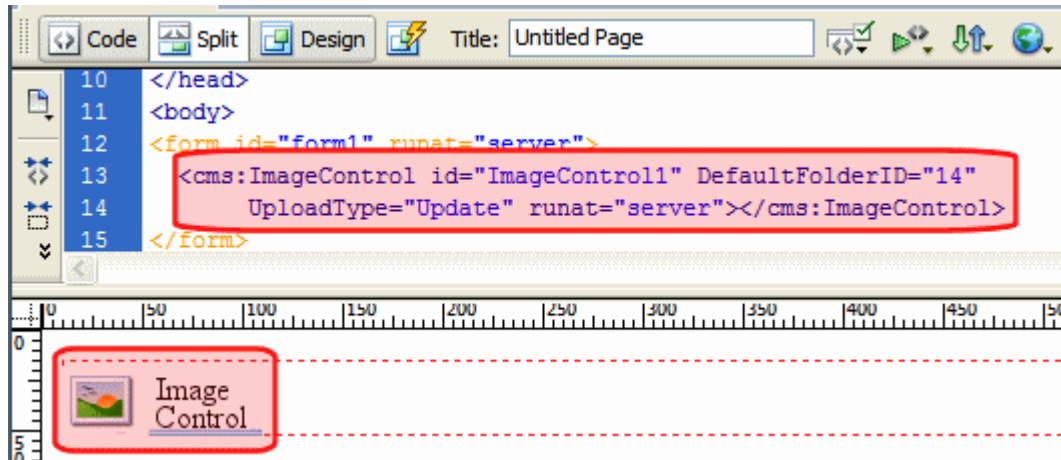
4. Use the following table to assist you with inserting the ImageControl function.

Field	Description
Control ID	A unique name for the function you are inserting.
FolderID	The ID of the folder where Images are located. This property is used when the UploadType property is set to Update. If you do not know the Folder ID, you can navigate to it using the Select Folder drop down box.
UploadType	Select whether the control adds new assets or update existing ones. Select Add to add assets and use the FolderID property. The default is <b>None</b> .
Hide	Used to hide a function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide function <input type="checkbox"/> <b>Unchecked</b> = Show function <hr/> <u>Note: If Hide and Visible are both checked, the property is hidden.</u>

Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

An ImageControl icon appears and indicates the ImageControl's location on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:ImageControl id="ImageControl1" DefaultFolderID="14" UploadType="Update" runat="server"&gt;&lt;/cms:ImageControl&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Language API

For a description of the Language API function, see ["LanguageAPI Server Control" on page 173](#).

This tag appears on the following menu(s):

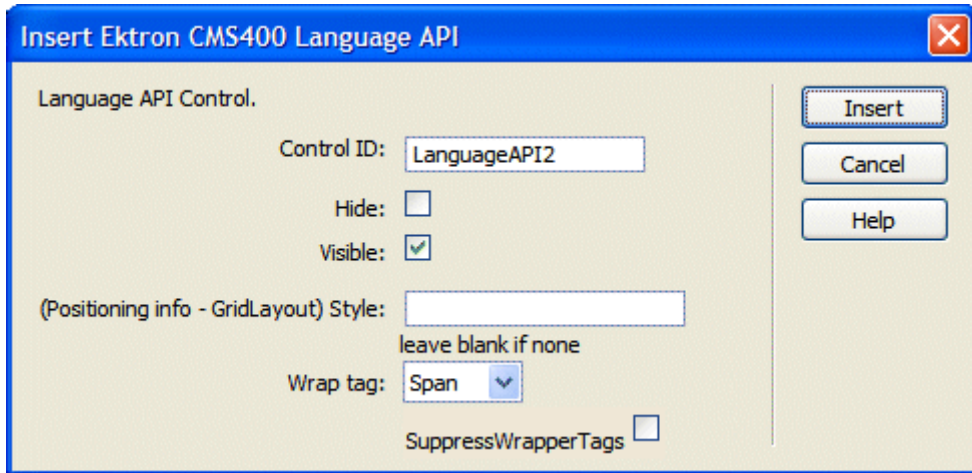
- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a Language API:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Language API**



3. The Language API box appears.



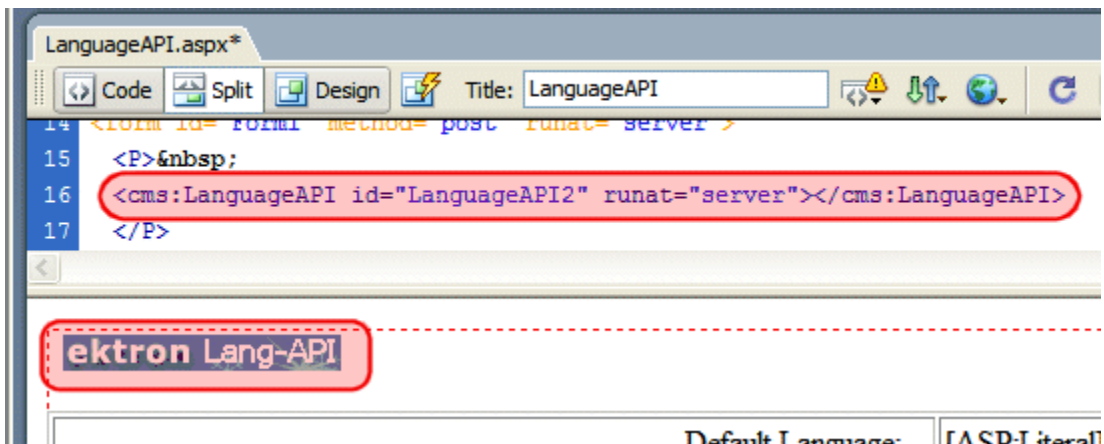
4. Use the following table to assist you with inserting a Language API box.

Field	Description
Control ID	A unique name for the function you are inserting.
Hide	<p>Used to hide a Language API in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Language API</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Language API</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>

Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. When completed, click **insert**.

A graphic defines where the Language API appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:LanguageAPI id="LanguageAPI2" runat="server"&gt;&lt;/cms:LanguageAPI&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.

Platform	Code View
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Language Select Box

For a description of the Language Select Box, see "[LanguageSelect Server Control](#)" on page 176.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

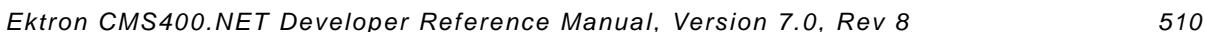
To insert a Language Select Box:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Language Select**
3. The Language Select box appears.

4. Use the following table to assist you with inserting a Language Select box.

Field	Description
Control ID	A unique name for the function you are inserting.
Hide	<p>Used to hide a function in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>

- When completed, click **insert**.  
A graphic defines where the Language Select appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:LanguageSelect id="LanguageSelect1" runat="server"&gt;&lt;/cms:LanguageSelect&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## List Summary

For a description of the List Summary function, see "[ListSummary Server Control](#)" on page 178.

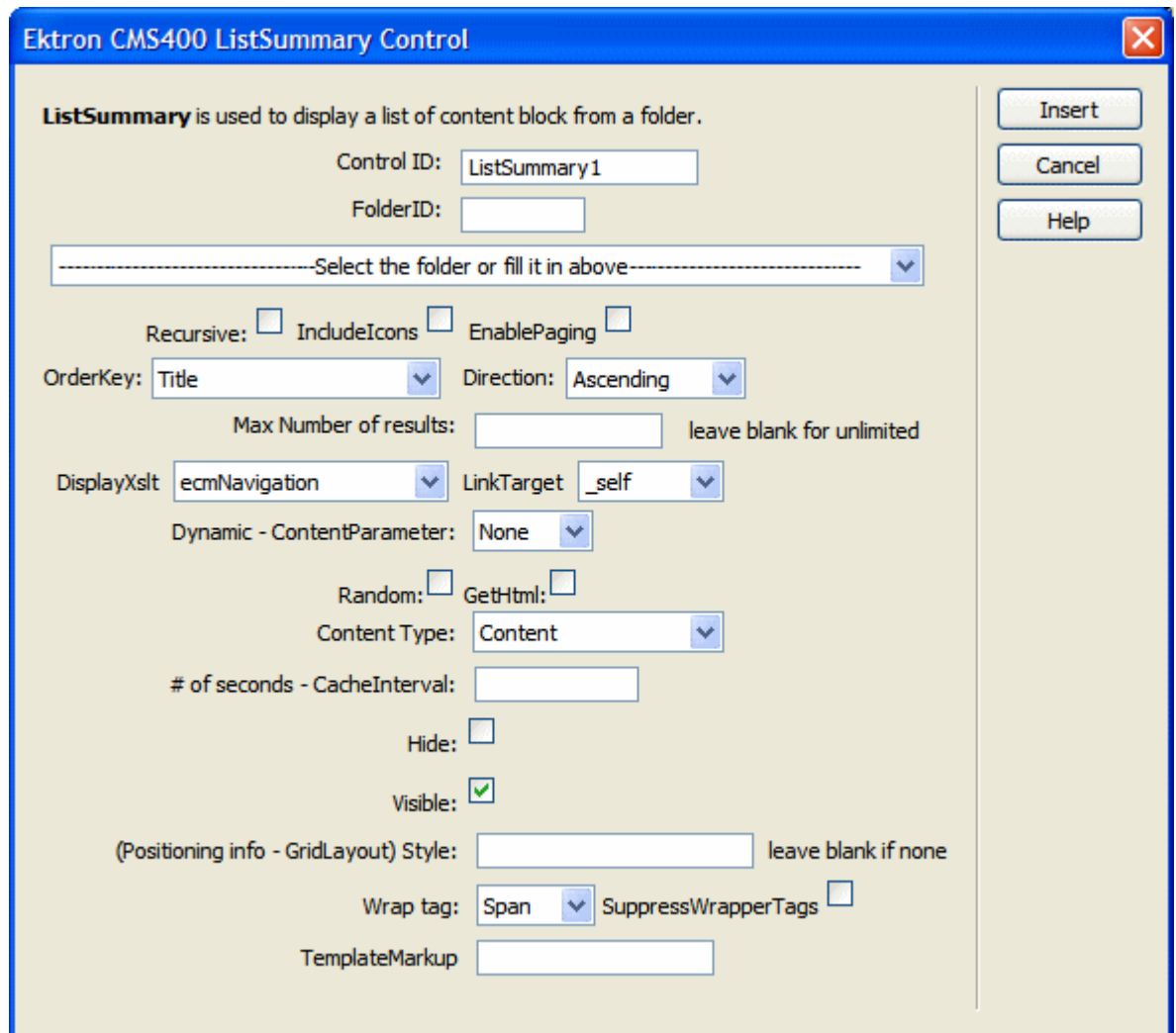
This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a list summary:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **List Summary**.

3. The Insert Ektron List Summary box is displayed.



**Ektron CMS400 ListSummary Control**

**ListSummary** is used to display a list of content block from a folder.

Control ID:

FolderID:

▼

Recursive: ☐ IncludeIcons ☐ EnablePaging ☐

OrderKey:  ▼ Direction:  ▼

Max Number of results:  leave blank for unlimited

DisplayXslt:  ▼ LinkTarget:  ▼

Dynamic - ContentParameter:  ▼

Random: ☐ GetHtml: ☐

Content Type:  ▼

# of seconds - CacheInterval:

Hide: ☐

Visible: ☒

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:  ▼ SuppressWrapperTags ☐

TemplateMarkup



Insert  
Cancel  
Help

**IMPORTANT!**


When inserting a list summary using ASP, CF, JSP or PHP, a different dialog box appears.

4. Use the following table to assist you with inserting a metadata control.

Field	Description
Control ID <hr/> <i>Used in ASP.NET only.</i> <hr/>	A unique name for the function you are inserting.
Folder ID	Enter the folder ID. If you do not know the folder path, you can navigate to it using the Select folder drop down box.
Select Folder	Select the folder the summary information comes from. The folder drop-down contains folders from your site. If you insert the folder path, you do not need to select a folder from the dropdown box.
Recursive	If checked, summary information is gathered from the folder selected and all of its child folders.

Field	Description
<p>Includelcons</p> <hr/> <p>Used in ASP.NET only.</p> <hr/>	<p>Choose whether to display icons next to the collection list's links.</p> <hr/> <p>Important: This property only works when the ecmSummary or ecmTeaser are used in the DisplayXslt property, or when the [\$Imagelcon] variable is used in an EkML file.</p> <hr/> <div> <a href="#">Ektron Rated Positive</a> Ektron Inc., an innovator in Web content management software, today announced ...  <a href="#">Ektron to Demonstrate Healthcare</a> "Healthcare Content Indexing Framework" which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.</div>



Field	Description
EnablePaging <hr/> Used in ASP.NET only. <hr/>	<p>This property, in conjunction with the <code>MaxResults</code> property, lets site visitors view an unlimited number of taxonomy items while controlling the amount of screen space. To accomplish this, the taxonomy display is limited to the number set in the <code>MaxResults</code> property.</p> <p>If you check this property box, and the number of taxonomy items exceeds the <code>MaxResults</code> number, navigation aids appear below the last item. The site visitor uses the aids to view additional items. See example below.</p>  <p>So, for example, if a taxonomy has 9 items and the <code>MaxResults</code> property is set to 3, the screen displays only the first three items. When the site visitor clicks <b>[Next]</b>, he sees items 4, 5 and 6, etc.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = enables paging</p> <p><input type="checkbox"/> <b>Unchecked</b> = disables paging</p> <hr/> <p><b>Important!</b> If the <code>EnablePaging</code> property is set to True, the <code>CacheInterval</code> property is disabled.</p> <hr/>
OrderKey	<p>Specify how the summaries are ordered. Sort by the following criteria.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>Last Author Last Name</b> - The last author's last name</li> <li>• <b>Last Author First Name</b> - The last author's first name</li> <li>• <b>Ascending</b> - Alphabetic or numerical order</li> <li>• <b>Descending</b> - Reverse alphabetic or numerical order</li> </ul>

Field	Description
Max Number of results <hr/> Used in ASP.NET only. <hr/>	<p>Enter the maximum number of items to appear in the initial display of this server control.</p> <p>To set no maximum, enter zero (0).</p> <p>To let site visitors view more than the maximum but limit the amount of space being occupied, enter the maximum number of results per page here. Then, check the <code>EnablePaging</code> property box.</p> <p>If you do and more than the number of <code>MaxResults</code> are available, navigation aids appear below the last item to help the site visitor view additional items. See example below.</p> <div data-bbox="718 556 1132 784" data-label="Image"> <p>The screenshot shows a list of four names, each on a new line and underlined: Bill Lumbergh, Bob Porter, Bob Slydell, and Milton Waddams. Below the list are four pagination controls: [First], [Previous], [Next], and [Last]. The [Previous] and [Next] controls are highlighted with a yellow background.</p> </div>
DisplayXslt <hr/> Used in ASP.NET only. <hr/>	<p>Determines how the information on the page is displayed. Select:</p> <p><b>None</b> - databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder  <i>See Also:</i> "Example of ecmNavigation Display" on page 101</p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary  <i>See Also:</i> "Example of ecmTeaser Display" on page 102</p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page</p> <hr/> <p><i>Note:</i> If an .ekml file is inserted into the MarkupLanguage property, this value is ignored.</p> <hr/>

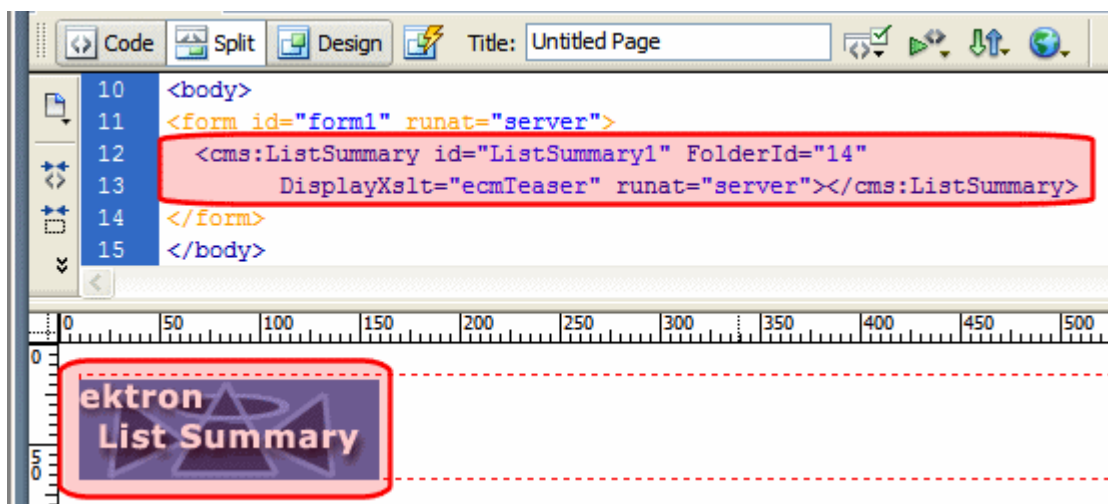
Field	Description
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "_self" if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "_self" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>
Dynamic Content Parameter <hr/> Used in ASP.NET only.	<p>Checks the QueryString for this value and replaces the list summary with a content block when specified. Choose <b>None</b> to always display List Summary.</p>
Random Output <hr/> Used in ASP.NET only.	<p>Displays random content within a list summary.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = display random content from list summary</p> <p><input type="checkbox"/> <b>Unchecked</b> = show a list summary</p>
GetHtml <hr/> Used in ASP.NET only.	<p>Check this box if you want to retrieve and display content (html body) for all content blocks in the collection. For example, to display content inside a web server control such as a GridView.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = retrieve and display content (html body) for all content blocks in the collection</p> <p><input type="checkbox"/> <b>Unchecked</b> = do not retrieve and display content (html body) for all content blocks in the collection</p>
Content Type <hr/> Used in ASP.NET only.	<p>Defines what type of list summary appears.</p> <p><b>Content</b> - Displays only content blocks in the folder.</p> <p><b>All Types</b> - Displays all content in the folder.</p> <p><b>Forms</b> - Displays only forms in the folder.</p> <p><b>Archive_Content</b> - Displays archived content from the folder.</p> <p><b>Archive_Forms</b> - Displays archived forms from the folder.</p>

Field	Description
# of seconds - CacheInterval <hr/> Used in ASP.NET only. <hr/>	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: "Caching with Server Controls" on page 42. <hr/> <b>Important!</b> If the EnablePaging property is set to True, the CacheInterval property is disabled. <hr/>
Hide <hr/> Used in ASP.NET only. <hr/>	Used to hide a list summary in design time and run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide list summary <input type="checkbox"/> <b>Unchecked</b> = Show list summary <hr/> Note: If Hide and Visible are both checked, the property is hidden. <hr/>
Visible <hr/> Used in ASP.NET only. <hr/>	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> Note: If Hide and Visible are both checked, the property is hidden. <hr/>
Style (Positioning Info - GridLayout) <hr/> Used in ASP.NET only. <hr/>	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag <hr/> Used in ASP.NET only. <hr/>	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

Field	Description
MarkupLanguage <hr/> Used in ASP.NET only.	<p>Identify the template markup file that controls the display of the list summary. For example, mylistsummary.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example, \cms400Min\workarea\customfiles\markup\mylistsummary.ekml.</p> <p>See Also: "Ektron Markup Language" on page 754 and "listsummary.ekml" on page 783</p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p>

- When completed, click **insert**.

A graphic defines where the list summary appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<pre>&lt;cms:ListSummary id="ListSummary1" FolderId="14" DisplayXslt="ecmTeaser" runat="server"&gt;&lt;/cms:ListSummary&gt;</pre>
ASP	<pre>&lt;% ecmListSummary "\Human Resources",1,1,"","DateModified,desc", "DateCreated",0,"","" %&gt;</pre>

Platform	Code View
ColdFusion	<pre>&lt;cfmodule template="#request.ecm.AppPath#ecmListSummary.cfm" Folder="\Human Resources" MaxNumber="0" Recursive="1" ShowSummary="1" StyleInfo="" OrderBy="DateModified,desc" ShowInfo="DateCreated"&gt;</pre>
JSP	<pre>&lt;%= ecmListSummary("\Human Resources",0,1,"", "Title,asc","DateModified",0,"","") %&gt;</pre>
PHP	<pre>&lt;?php ecmListSummary("\Human Resources",1,1,"", "DateModified,desc","DateCreated",0,"",""); ?&gt;</pre>

## Login

For a description of the Login button, see ["Login Server Control" on page 186](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- **Commands > Ektron CMS400.NET (Server Controls) > CMS Content Controls...**

To insert a login button:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Login**.

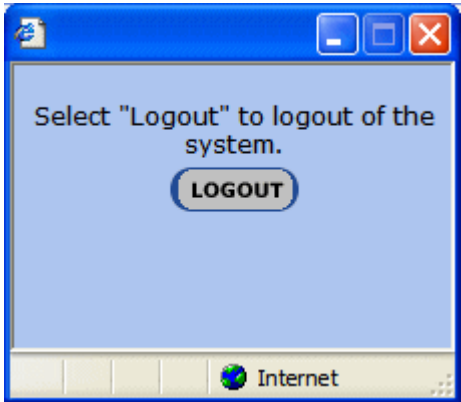
3. The Insert Ektron CMS400.NET Login box appears.

**IMPORTANT!**




When inserting a login function using ASP, CF, JSP or PHP, the dialog box that appears has no settings.

4. Use the following table to assist you with inserting a Login button.

Field	Description
ID <hr/> Used in ASP.NET only.	A unique name for the function you are inserting.

Field	Description
OnlyAllowMemberLogin <hr/> Used in ASP.NET only. <hr/>	<p>This changes the Login button so only membership users can log in. This prevents someone from trying to login as an Ektron CMS400.NET user and gain access to the Ektron CMS400.NET Workarea. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Allow only membership users to login.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Allow Ektron CMS400.NET users and membership users to login.</p>
PromptLogout <hr/> Used in ASP.NET only. <hr/>	<p>When set to False, the logout process will not include the Logout Prompt window (shown below).</p>  <p><b>True</b> = Users must click the Logout button in the Logout prompt window before they are logged out.</p> <p><b>False</b> = Users are logged out without having to use the Logout prompt window.</p>
AutoLogin <hr/> Used in ASP.NET only. <hr/>	<p>When this box is checked, users are automatically logged in using Active Directory authentication. They do not need to enter a username or password. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Use Active Directory authentication when logging in.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not use Active Directory authentication when logging in.</p> <hr/> <p><b>Important!:</b> For this property to function properly, you must be using Active Directory authentication with your CMS400.NET Site.</p> <hr/>

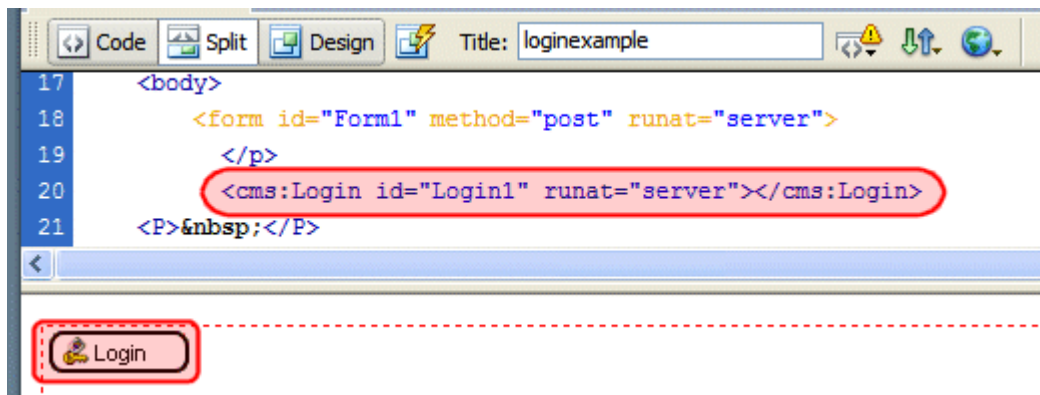


Field	Description
AutoAddType <hr/> Used in ASP.NET only. <hr/>	Set this type to define automatically added users as membership users or CMS400.NET content authors. <b>Author</b> = CMS400.NET content authors <b>Member</b> = membership users
SuppressHelpButton <hr/> Used in ASP.NET only. <hr/>	Hides the Help button that appears below the Login button when this box is checked. The default is Unchecked. <input checked="" type="checkbox"/> <b>Checked</b> = Do not display the Help button.  <input type="checkbox"/> <b>Unchecked</b> = Display Help button.  
Hide <hr/> Used in ASP.NET only. <hr/>	Used to hide a Login button at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide Login button <input type="checkbox"/> <b>Unchecked</b> = Show Login button <hr/> Note: If Hide and Visible are both checked, the property is hidden.
Visible <hr/> Used in ASP.NET only. <hr/>	Used to show or hide the function at run time. <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function <hr/> Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout) <hr/> Used in ASP.NET only. <hr/>	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"

Field	Description
Wrap Tag <hr/> Used in ASP.NET only.	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper <hr/> Used in ASP.NET only.	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

5. Click **insert**.

A login icon indicates where the Login button appears on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Login id="Login1" runat="server"&gt;&lt;/cms:Login&gt;</code>
ASP	<code>&lt;% ecmlogin %&gt;</code>
ColdFusion	<code>&lt;cfmodule template="#request.ecm.AppPath#ecmLogin.cfm"&gt;</code>
JSP	<code>&lt;%= ecmLogin() %&gt;</code>
PHP	<code>&lt;?php ecmlogin() ?&gt;</code>

## Map Control

For a description of the Map Control, see ["Map Server Control" on page 190](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS Module Controls...**

To insert a Map Control button:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Map Control**.
3. The Ektron Map Control dialog box appears.

**Ektron MAP**

**MAP**

Control ID:

FolderID:  Recursive ☒

-----Select the folder or fill it in above-----

Longitude  Latitude

MyAddress  MapCategory

MinZoomLevel  StartZoomLevel

MapStyle  PageSize

MapProvider  DistanceUnit

CustomIcon ☐ DisplayTextResult ☒

TypeControl ☒ ZoomControl ☒ GeoControl ☒

TemplateMarkup  Hide ☐ Visible: ☒

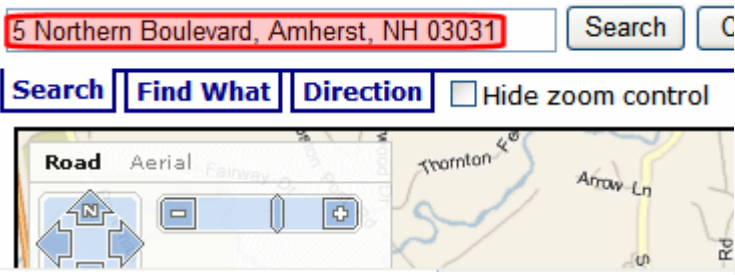
(Positioning info - GridLayout) Style:  leave blank if none

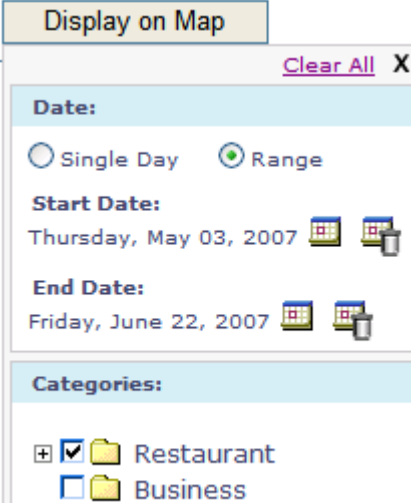
Wrap tag:  SuppressWrapperTags ☐

Insert Cancel Help

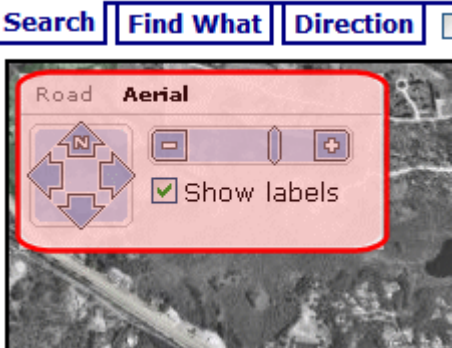
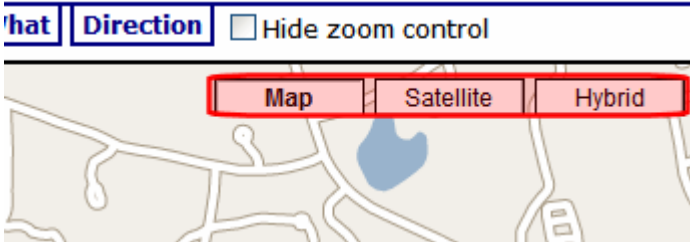
4. Use the following table to assist you with inserting a Map Control.

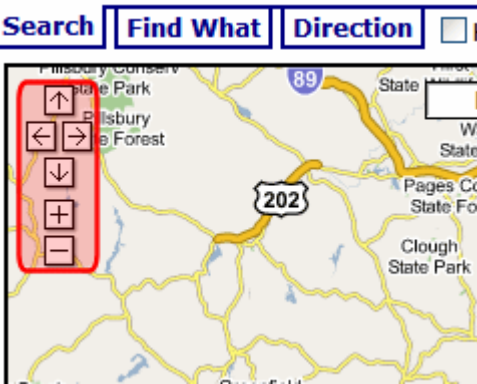
Field	Description
Control ID	A unique name for the function you are inserting.

Field	Description
Folder ID	Enter the folder ID. If you do not know the folder path, you can navigate to it using the Select folder drop down box.
Recursive	In the <b>FolderID</b> property, you specify a folder whose content is searched on this map. To extend the search to all folders below this folder, check this box.
Latitude	To set a default map center via latitude and longitude (as opposed to an address), enter the latitude here.
Longitude	To set a default map center via latitude and longitude (as opposed to an address), enter the longitude here.
MyAddress	<p>To set a default map center, enter its address here. If you only enter a zip code, the map centers on its post office.</p> <p>The address appears in the <b>Search</b> field when the map first displays.</p>  <p>If you enter an address, the latitude and longitude properties are ignored.</p>

Field	Description
MapCategory	<p>Enter the ID number of the taxonomy whose categories appear when a site visitor clicks this map's <b>Display on Map</b> button.</p> <p>When a map first appears, all eligible content appears. If a site visitor clicks the <b>Display on Map</b> button, he can choose one or more categories and limit the map to items assigned to them. For example, the user could view restaurants only (as shown below).</p> <p>As another example, your map could display all campuses in your state's college system. A site visitor could use the <b>Display on Map</b> popup window to limit the map to community colleges.</p> <p>To learn more about Taxonomy, see the <b>Ektron CMS400.NET</b> Administrator Manual chapter "Taxonomy."</p> 
MinZoomLevel	<p>If you want to set a map zoom level below which map locations will not appear, enter that value.</p> <p>The default value is 4. Possible values are between 1 (most detailed) and 19.</p>
StartZoomLevel	<p>Enter the zoom level at which the map initially appears. See <i>Also:</i> <a href="#">"Determining a Map's Initial Boundaries" on page 201</a></p>
MapStyle	<p>Enter the map's display mode: Road, Satellite or Hybrid.</p> <hr/> <p><a href="#">Note: This setting only affects Virtual Earth maps.</a></p> <hr/>

Field	Description
PageSize	<p>Enter the number of locations that can appear on one page of the text box after a search is executed. See Also: "<a href="#">Displaying/Suppressing Map Elements</a>" on page 200</p> <p>If more than this number of locations are available, the site visitor sees <b>[First]</b> <b>[Previous]</b> <b>[Next]</b> <b>[Last]</b> at the bottom of the page. He can use the navigation aids to view additional locations.</p>
MapProvider	Select the service that provides the map, either Google or Virtual Earth.
DistanceUnit	Enter the units of distance on the map. Choices are miles and kilometers. The default value is miles.
CustomIcon	<p>Use this field if you want the Category popup box to display an icon to the left of each category, as shown below.</p> <p><b>Prerequisite</b></p> <p>To use this value, open the <code>webroot/images/application/maps/tree</code> folder. In that folder, create a new folder whose name is the same as the Taxonomy category assigned to the Map server control at the MapCategory property. In the sample site, this Taxonomy's name is MapCategory.</p> <p>Then, place the icons in that folder by category title name with a .png extension. Use an underscore ( <code>_</code> ) to separate taxonomy levels. For example, the image for the category Restaurant &gt; American must be named <code>Restaurant_American.png</code>.</p>

Field	Description
DisplayTextResult	<p>If you want the site visitor to see a box of information about each map item to the right of a map, check this box. To suppress the text box, uncheck this box. See Also: <a href="#">"Displaying/Suppressing Map Elements" on page 200</a></p> <p><input checked="" type="checkbox"/> <b>Checked</b> = display text results about each map item</p> <p><input type="checkbox"/> <b>Unchecked</b> = suppress text box</p>
TypeControl	<p>If this map uses Virtual Earth maps, this property enables or disables the zoom/direction/type control (highlighted below). It lets a site visitor zoom the map in and out, move the center in any direction, and change the display style (Road or Aerial).</p>  <p>If this map uses Google Earth maps, this property enables or disables the type control (highlighted below). It lets the site visitor change the display style (Map, Satellite, or Hybrid). Use the ZoomControl property to display or suppress Google Earth's zoom and direction controls.</p>  <p><input checked="" type="checkbox"/> <b>Checked</b> = display zoom/direction/type controls</p> <p><input type="checkbox"/> <b>Unchecked</b> = suppress zoom/direction/type controls</p>

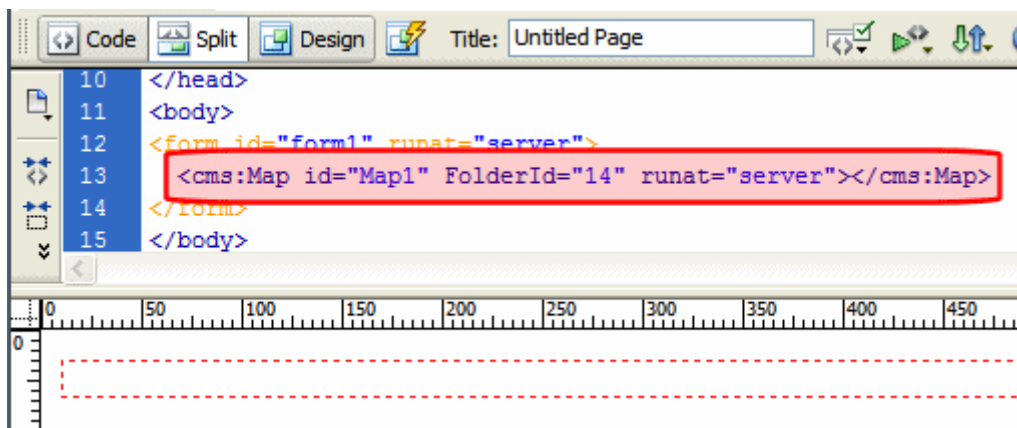
Field	Description
ZoomControl	<p>Use this property to display or suppress the Google map zoom control (highlighted below).</p>  <p>For Virtual Earth maps, the display of the zoom control is determined by the <b>TypeControl</b> property.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = display Google map zoom control</p> <p><input type="checkbox"/> <b>Unchecked</b> = suppress Google map zoom control</p>
Hide	<p>Used to hide a Login button at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Login button</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Login button</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>



Field	Description
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

A Map Control icon indicates where the Map Control appears on the template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Map id="Map1" FolderId="14" runat="server"&gt;&lt;/cms:Map&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.

Platform	Code View
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Membership Control

For a description of the Membership function, see ["Membership Server Control" on page 214](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls...**

To insert a Membership Control:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Membership**.

3. The Membership dialog box appears.

**Ektron CMS400 membership**

Membership Control

ID:

DisplayMode:  ▼

RedirectFailedURL:

RedirectSuccessURL:

RegisterButtonImg:

RegisterButtonText:

ResetButtonImg:

ResetButtonText:

ShowExtended: ☒ ShowTerms ☐

UserExistsMessage:

UserSuccessMessage:

UserUpdateSuccessMessage:

BoardID:

Hide: ☐ Visible: ☒

(Positioning info - GridLayout) Style:

Wrap tag:  ▼ SuppressWrapperTags ☐

Insert  
Cancel  
Help

4. Use the following table to assist you with inserting a Membership Control.

Field	Description
ID	A unique name for the function you are inserting.

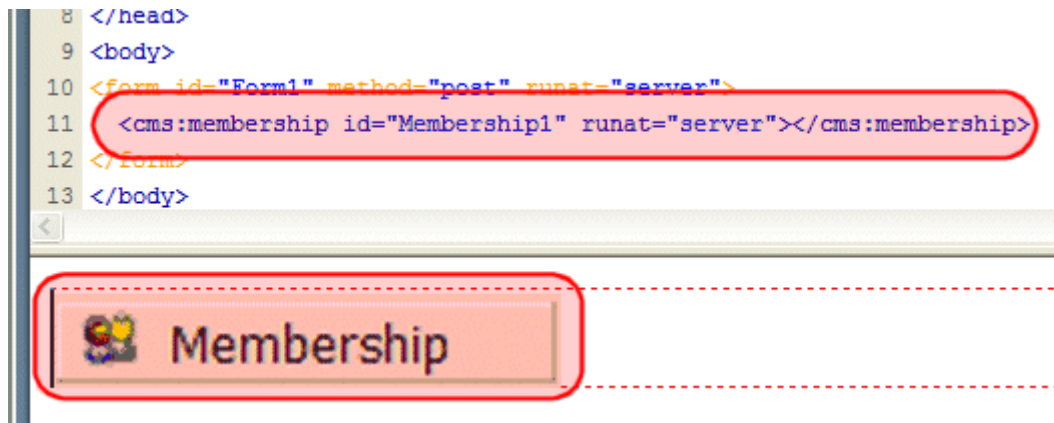
Field	Description
DisplayMode	<p>The type of membership form displayed. Choose from the following:</p> <ul style="list-style-type: none"> <li>• <b>UserRegistration</b> - Lets a site visitor to register as a membership user.</li> <li>• <b>ResetPassword</b> - Lets a membership user to reset his password.</li> <li>• <b>UnsubscribeSecured</b> - Lets a membership user unsubscribe. The membership user needs to enter his username and password to unsubscribe.</li> <li>• <b>UnsubscribeUnsecured</b> - Lets a membership user unsubscribe by entering his membership username only.</li> <li>• <b>AccountActivate</b> - Let a site visitor activate his membership account by entering the ID number sent him in the account verification email.</li> </ul> <p>For more information on the DisplayMode property, see <a href="#">"DisplayMode" on page 219</a>.</p>
RedirectedFailedURL	<p>The URL where a membership user is sent if the registration fails. If the page resides in the same folder as the registration page, you only need to enter the name of the page. For example, RegFailed.aspx. If the redirect page is located in a subfolder, you need to add the folder path. For example, members\RegFailed.aspx.</p>
RedirectedSuccessURL	<p>The URL where a membership user is sent if the registration fails. If the page resides in the same folder as the registration page, you only need to enter the name of the page. For example, RegFailed.aspx. If the redirect page is located in a subfolder, you need to add the folder path. For example, members\RegFailed.aspx.</p>
RegisterButtonImg	<p>Lets you add an image in place of the text on the register button.</p>
RegisterButtonText	<p>Text that appears on the Register button. The default is Register. If you use a register button image, you do not see this text.</p>
ResetButtonImg	<p>Lets you add an image in place of the text on the reset button.</p>
ResetButtonText	<p>Text that appears on the reset button. The default is Reset. If you use a reset button image, you do not see this text.</p>

Field	Description
ShowExtended	<p>Decide if the Custom User Properties are available when using this control. The default setting is checked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = show Custom User Properties</p> <p><input type="checkbox"/> <b>Unchecked</b> = hide Custom User Properties</p> <hr/> <p><b>Note:</b> For more information on Custom User Properties, see the <a href="#">Administration Manual</a> section "Custom User Properties".</p> <hr/>
ShowTerms	<p>When set to True this property shows the Terms and Conditions as defined in the Workarea. In addition, users are asked to check a box stating they have read and agree to abide by the terms and conditions of the forum. Default is <b>False</b>.</p> <p>See Also: "<a href="#">ShowTerms</a>" on <a href="#">page 221</a> and the Administrator Manual section "Discussion Boards" &gt; "Implementing Discussion Boards" &gt; "Working with Discussion Boards and Categories" &gt; "Terms &amp; Conditions"</p>
UserExistsMessage	The message that appears when a membership user already exists. The default message is: Username(email) already exists!
UserSuccessMessage	The message that appears when a membership user successfully registers. The default message is: You have registered successfully.
UserUpdateSuccessMessage	The message that appears when a membership user successfully updates their information. The default message is: You have successfully updated your information.
Hide	<p>Used to hide a Membership in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Membership</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Membership</p> <hr/> <p><b>Note:</b> If Hide and Visible are both checked, the property is hidden.</p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><b>Note:</b> If Hide and Visible are both checked, the property is hidden.</p> <hr/>

Field	Description
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

5. When completed, click **insert**.

A graphic defines where the Membership Control appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:membership id="Membership1" runat="server"&gt;&lt;/cms:membership&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.

Platform	Code View
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Metadata for ASP.NET Pages

For a description of the Metadata function, see ["MetaData Server Control" on page 261](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a metadata function:

1. Move the blinking cursor to the area that you want to place it.

**NOTE** The Ektron CMS400.NET Metadata function should be inserted within the <head> tags of your template to work properly.

2. From the list of Ektron CMS400.NET Functions, click on **Metadata**.
3. The Insert Ektron Metadata box is displayed.

**Insert Ektron CMS400 metadata control**

**MetaData**

Control ID:

Content ID:

GenerateDublinCore: ☐

-----Select the category to retrieve list of content-----

-----Select the content block to insert into the page-----

DefaultItemList

Dynamic Content Parameter:

Hide: ☐ Visible: ☒

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:   SuppressWrapperTags ☐

## 4. Use the following table to assist you with inserting metadata.

Field	Description
Control ID	A unique name for the function you are inserting.
Content ID	Enter a valid content ID in the Content ID textbox. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content Block drop down boxes.
GenerateDublinCore	<p>When enabled, this property automatically creates seven of the Simple Dublin Core metadata fields from standard CMS400.NET system properties. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Generate Simple Dublin Core metadata fields</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not generate Simple Dublin Core metadata fields</p> <p>The seven fields and how they are associated with the CMS400.NET properties is explained in <a href="#">"Using the Simple Dublin Core Metadata Standard" on page 263</a></p>
Select Category	The drop-down list contains a list of content folders. If you insert the content ID, you do not need to select a category.
Select Content Block	The drop-down box is populated with a list of all the content block titles within that folder. When a title is selected, the content ID textbox is filled with that content block's id number. If you insert the content ID, you do not need to select a content block.
DefaultItemList	A comma delimited list of content block IDs to get metadata from. DefaultContentID must be set to 0 (zero) so you can add IDs to the list. For example: <b>[2,4,46]</b>
Dynamic Content Parameter	Gets or sets the QueryString parameter to read a content ID dynamically.
Hide	<p>Used to hide a Membership in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Membership</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Membership</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>



Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. When completed, click **insert**.

6. The following line appears in the <body> of the HTML code.

```
<cms:MetaData id="MetaData2" runat="server" DefaultContentID="12"></cms:MetaData>
```

7. Move that line from the <body> tag into the <head> tag.

```
<HEAD>
 <title>WebForm5</title>
 <cms:MetaData id="Metadadata2" runat="server" DefaultContentID="12"></cms:MetaData>
 <meta name="GENERATOR" content="Microsoft Visual Studio .NET 7.1">
 <meta name="CODE_LANGUAGE" content="Visual Basic .NET 7.1">
 <meta name="vs_defaultClientScript" content="JavaScript">
 <meta name="vs_targetSchema" content="http://schemas.microsoft.com/intellisense/1
</HEAD>
<body>
 <form id="Form1" method="post" runat="server">|
 </form>
</body>
```

8. Save the Web form.
9. Open the Web page in the browser.
10. Right click on the Web page and click **View Source**. Look in the head tag.

The meta tags from the content block are added to the page.

The metadata information added is shown below.

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
 <HEAD>
 <title>webForm5</title>
 <title>CMS Developer</title>
 <meta name="keywords" content="CMS400; .NET; CMS">
 <meta name="description" content="CMS400 developer section">
 <meta content="Microsoft Visual Studio .NET 7.1" name="GENERATOR">
 <meta content="Visual Basic .NET 7.1" name="CODE_LANGUAGE">
 <meta content="JavaScript" name="vs_defaultClientScript">
 <meta content="http://schemas.microsoft.com/intellisense/ie5"
name="vs_targetSchema">
 </HEAD>

```

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:MetaData id="MetaData2" runat="server" DefaultContentID="12"></cms:MetaData>
ASP	For ASP pages, see <a href="#">"Metadata Function for ASP, CF, JSP or PHP Pages" on page 540</a>
ColdFusion	For ColdFusion pages, see <a href="#">"Metadata Function for ASP, CF, JSP or PHP Pages" on page 540</a>
JSP	For JSP pages, see <a href="#">"Metadata Function for ASP, CF, JSP or PHP Pages" on page 540</a>
PHP	For PHP pages, see <a href="#">"Metadata Function for ASP, CF, JSP or PHP Pages" on page 540</a>

## Metadata Function for ASP, CF, JSP or PHP Pages

For a description of the Metadata function, see ["MetaData Server Control" on page 261](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a metadata function:

1. Move the blinking cursor to the area that you want to place it.

**NOTE** The Ektron CMS400.NET Metadata function should be inserted within the <head> tags of your template to work properly.

2. From the list of Ektron CMS400.NET Functions, click on **Metadata**.
3. The Insert Ektron Metadata box is displayed.

4. Use the following table to assist you with inserting metadata.

Field	Description
Metadata List	When you select a content ID and click the Add to Metadata List button, The content ID is added to this text box. This allows you to create a list of content blocks for which the metadata of each content block is added to the page.
Content ID	Enter a valid content ID in the Content ID textbox. If you do not know the Content ID, you can navigate to it using the Select Category and Select Content Block drop down boxes.

Field	Description
GenerateDublinCore	<p>When enabled, this property automatically creates seven of the Simple Dublin Core metadata fields from standard CMS400.NET system properties. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Generate Simple Dublin Core metadata fields</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not generate Simple Dublin Core metadata fields</p> <p>The seven fields and how they are associated with the CMS400.NET properties is explained in <a href="#">"Using the Simple Dublin Core Metadata Standard" on page 263</a></p>
Select Category	The drop-down list contains a list of content folders. If you insert the content ID, you do not need to select a category.
Select Content Block	The drop-down box is populated with a list of all the content block titles within that folder. When a title is selected, the content ID textbox is filled with that content block's id number. If you insert the content ID, you do not need to select a content block.
Metadata types for exclusion	Select the metadata types that you do not want added to the metadata function you are inserting. Hold the control key down for multiple selections.

5. When completed, click **insert**.

6. The following line appears in the <body> of the HTML code.

```
<cms:MetaData id="MetaData2" runat="server" DefaultContentID="12"></cms:MetaData>
```

7. Move that line from the <body> tag into the <head> tag.

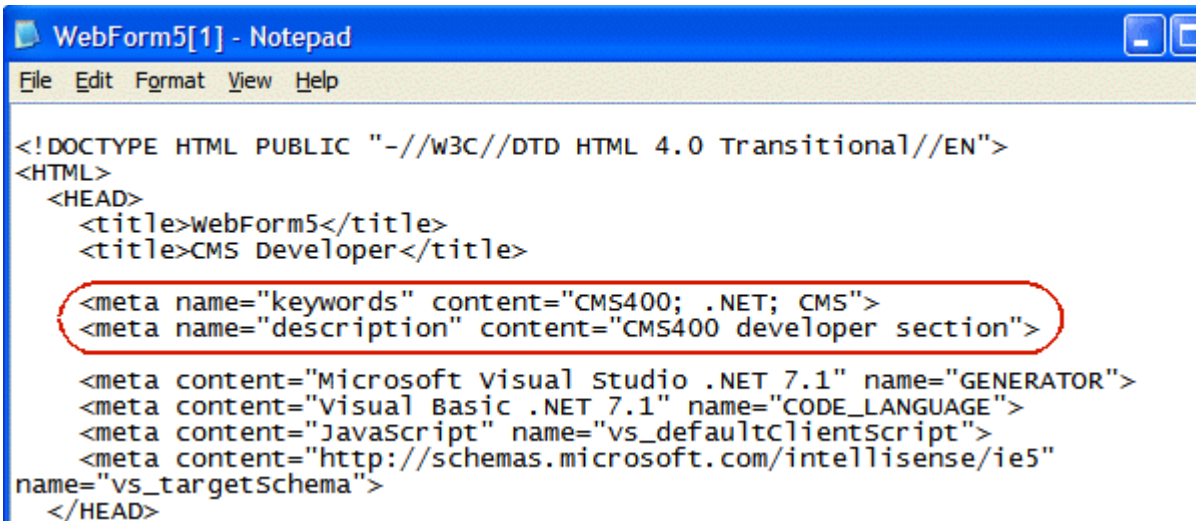
```
<HEAD>
<title>WebForm5</title>
<cms:MetaData id="Metadadata2" runat="server" DefaultContentID="12"></cms:MetaData>
<meta name="GENERATOR" content="Microsoft Visual Studio .NET 7.1">
<meta name="CODE_LANGUAGE" content="Visual Basic .NET 7.1">
<meta name="vs_defaultClientScript" content="JavaScript">
<meta name="vs_targetSchema" content="http://schemas.microsoft.com/intellisense/...>
</HEAD>
<body>
<form id="Form1" method="post" runat="server">|
</form>
</body>
```

8. Save the Web form.

9. Open the Web page in the browser.

10. Right click on the Web page and click **View Source**. Look in the head tag. The meta tags from the content block are added to the page.

The metadata information added is shown below.



```

WebForm5[1] - Notepad
File Edit Format View Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
 <HEAD>
 <title>webForm5</title>
 <title>CMS Developer</title>
 <meta name="keywords" content="CMS400; .NET; CMS">
 <meta name="description" content="CMS400 developer section">
 <meta content="Microsoft Visual Studio .NET 7.1" name="GENERATOR">
 <meta content="Visual Basic .NET 7.1" name="CODE_LANGUAGE">
 <meta content="JavaScript" name="vs_defaultclientscript">
 <meta content="http://schemas.microsoft.com/intellisense/ie5"
name="vs_targetschema">
 </HEAD>

```

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	For ASP.NET pages, see <a href="#">"Metadata for ASP.NET Pages" on page 537</a>
ASP	<% ecmMetadata "[1;Title][9][3;Keywords,Title]","" %>
ColdFusion	<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm" ItemList="[1;Title][9][3;Keywords,Title]">
JSP	<%= ecmMetadata("[12;keywords,Title]","") %>
PHP	<?php ecmMetadata("[1;Title][9][3;Keywords,Title]",""); ?>

## Multipurpose Content Block

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a Multipurpose content block function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Multipurpose Tag...**

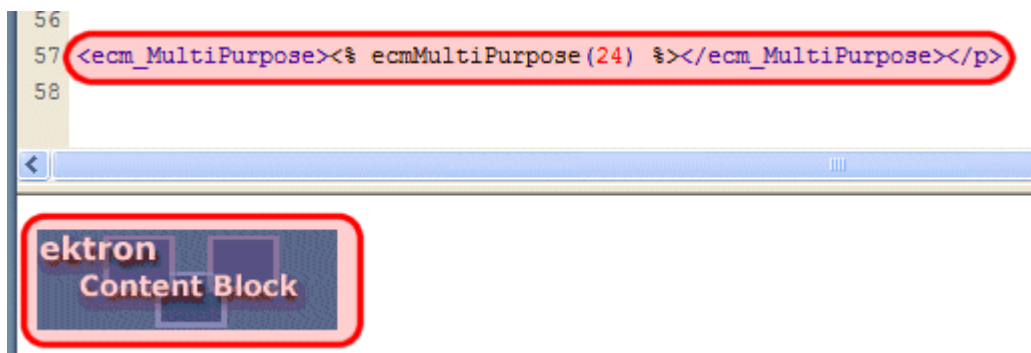
3. The Insert Ektron CMS400.NET Multipurpose Tag box is displayed.

4. Use the following table to assist you with inserting a Multipurpose tag.

Field	Description
Default Content ID	Type in the default content ID, or select a category from the drop down list, then select a content block from the second dropdown list.

5. When completed, click **insert**.

A graphic to define where the Multipurpose tag will appear on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
.NET	In ASP.NET, the Content Block Tag functions as a Multipurpose Content block.
ASP	<code>&lt;% ecmMultiPurpose(24) %&gt;</code>
ColdFusion	<code>&lt;cfmodule DefaultID="24" template="#request.ecm.AppPath#ecmMultipurpose.cfm"&gt;</code>

Platform	Code View
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	<code>&lt;?php ecmMultiPurpose(12); ?&gt;</code>

## Poll

For a description of the Poll function, see ["Poll Server Control" on page 298](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a Poll:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click **Poll**
3. The Insert Poll box is displayed.

4. Use the following table to assist you with inserting a poll.

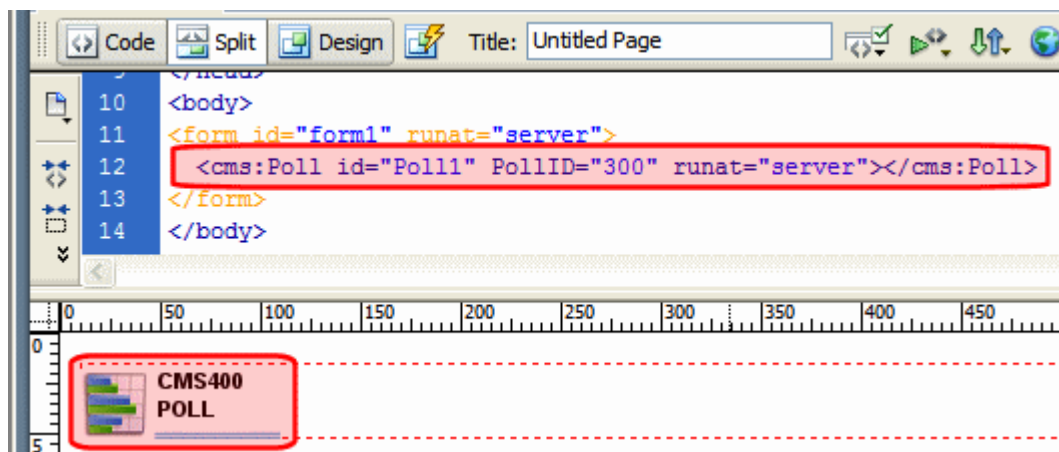
Field	Description
Control ID	A unique name for the function you are inserting.

Field	Description
Poll ID	Type in the Poll ID that you want to insert. If you do not know the Poll ID, you can navigate to it using the Select Poll drop down box.
Select Poll	The drop-down box is populated with a list of all the form titles. When a title is selected, the Poll ID textbox is filled with that form's id number. If you insert the Poll ID, you do not need to select a form.
Dynamic Form Parameter	To make this form block dynamic, select <b>ekfrm</b> . When you do, this control uses the form block passed as a URL parameter.
Ajax	<p>Displays the poll or results, using an iFrame, in the area of the page that contains the poll without disturbing its surrounding contents. The default is checked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Polls and results are shown in an iFrame without any modification or refresh of the surrounding contents.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The poll and the results replace the Web page's content.</p>
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the CacheInterval property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42.</a>
Include Form Tags	Determines if tags are generated automatically or manually. See Also: <a href="#">"Automatic versus Manual Generation of Form Tags" on page 161</a>
Add Validation	<p>When <b>checked</b>, adds Validation to your main runat=server form.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = add validation</p> <p><input type="checkbox"/> <b>Unchecked</b> = do not add validation</p>
Hide	<p>Used to hide a poll in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide poll</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show poll</p> <hr/> <p><b>Note:</b> If Hide and Visible are both checked, the property is hidden.</p> <hr/>



Field	Description
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

- When completed, click **Insert**.  
A graphic defines where the form appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:Poll id="Poll1" PollID="300" runat="server"&gt;&lt;/cms:Poll&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## PostHistory

For a description of the PostHistory function, see "[PostHistory Server Control](#)" on [page 157](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Community Controls... > Forums**

To insert the PostHistory function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **PostHistory**.

3. The Insert Ektron CMS400.NET PostHistory box appears.

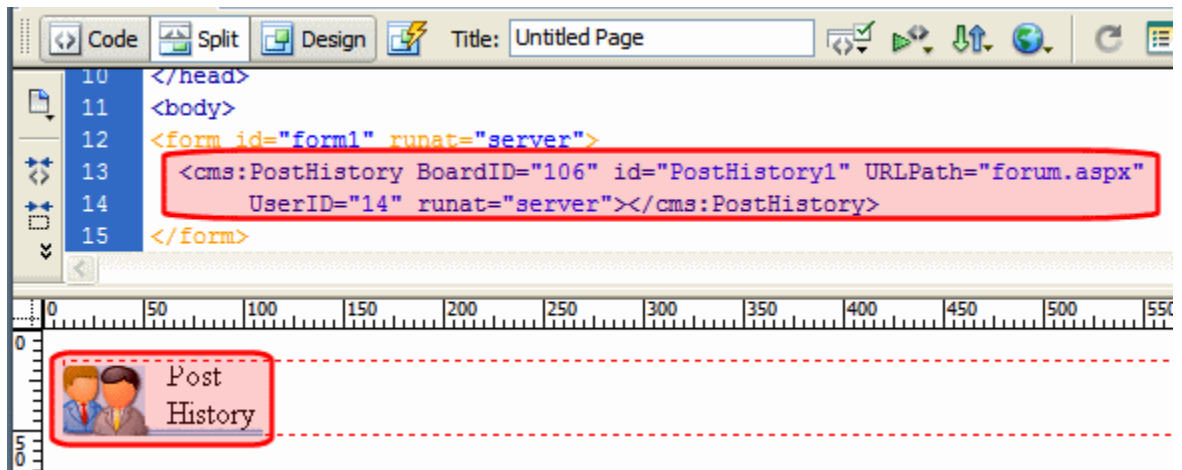
4. Use the following table to assist you with inserting the ActiveTopics function.

Field	Description
Control ID	A unique name for the function you are inserting.
BoardID	The ID of the discussion board. If you do not know the board ID, you can navigate to it using the Select folder drop down box.
UserID	The ID of the user for whom to get the post history.
URLPath	The URL path to the page the where the Forum server control is located.
MaxNumber	The maximum number of topics listed.
# of seconds - CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a>

Field	Description
Hide	<p>Used to hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><u>Note: If Hide and Visible are both checked, the property is hidden.</u></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

5. Click **insert**.

An PostHistory icon appears to indicate the location of the post history list.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:PostHistory BoardID="106" id="PostHistory1" URLPath="forum.aspx" UserID="14" runat="server"&gt;&lt;/cms:PostHistory&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Random Content

For a description of the Random Content function, see ["Random Content" on page 669](#)

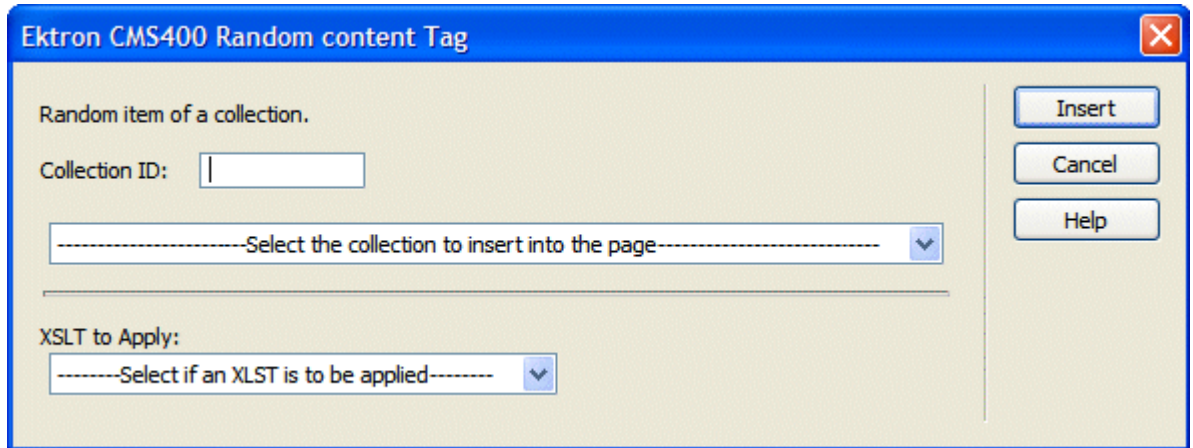
This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a Random Content function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Insert Random Content Tag...**

3. The Insert Random Content Tag box is displayed.



The dialog box is titled "Ektron CMS400 Random content Tag". It contains the following fields and controls:

- Random item of a collection.** (Label)
- Collection ID:** A text input field.
- A dropdown menu with the text "-----Select the collection to insert into the page-----".
- XSLT to Apply:** A dropdown menu with the text "-----Select if an XSLT is to be applied-----".
- Buttons: **Insert**, **Cancel**, and **Help**.

4. Use the following table to assist you with inserting a Random Content tag.

Field	Description
Collection ID	Type in the collection ID, or select it from the drop down list, of the collection where the random content function will retrieve content from.
XSLT to apply <i>XML Content only</i>	If desired, you may specify an XSLT to apply to the content that gets displayed.

5. When completed, click **Insert**.

A graphic to define where the Random Content tag will appear on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
.NET	This function is not available as an Ektron Dreamweaver Extension for .NET
ASP	<code>&lt;% ShowRandomContent 3, 0 %&gt;</code>
ColdFusion	<code>&lt;cfmodule id="3" displayFunction="0" template="#request.ecm.AppPath#ecmShowRandomContent.cfm"&gt;</code>
JSP	<code>&lt;%= ShowRandomContent(3, 0) %&gt;</code>
PHP	<code>&lt;?php ShowRandomContent(3, 0); ?&gt;</code>

## Random Summary

For a description of the Random Summary function, see ["Random Summary" on page 670](#)

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a Random Summary function:

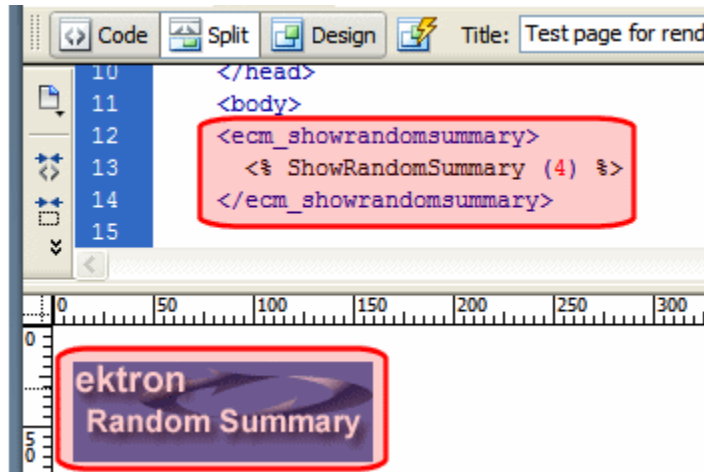
1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Random Summary Tag...**
3. The Insert Ektron CMS400.NET Random Summary Tag box is displayed.

4. Use the following table to assist you with inserting a Random Summary tag.

Field	Description
Collection ID	Type in the collection ID, or select it from the drop down list, of the collection where the random summary function will retrieve summaries from.

5. When completed, click **Insert**.

A graphic to define where the Random Summary tag will appear on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
.NET	This function is not available as an Ektron Dreamweaver Extension for .NET
ASP	<% ShowRandomSummary (2) %>
ColdFusion	<cfmodule id="2" template="#request.ecm.AppPath#ecmShowRandomSummary.cfm">
JSP	<%= ShowRandomSummary(2) %>
PHP	<?php ShowRandomSummary (2); ?>

## RSS Aggregator

For a description of the RSS Aggregator function, see ["RssAggregator Server Control" on page 302](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS Module Control...**

To insert an RSS Aggregator:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **RSS Aggregator**.



3. The RSS Aggregator dialog box appears.

4. Use the following table to assist you with inserting an RSS Aggregator.

Field	Description
Control ID	A unique name for the function you are inserting.
URL	The RSS feed path for the server control. For example: <code>http://example.com/rss.xml</code>
MaxResults	The Maximum number of items from an RSS feed that are returned (0=unlimited).
LinkTarget	Defines the way a link acts when a link is clicked. The choices are:  <b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.  <b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.  <b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.  <b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.

Field	Description
Display Type	<p>Determines how the information on the page is displayed</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every RSS feed item  See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists a title and a description of every RSS feed item  See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p> <p><b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page</p>
Hide	<p>Used to hide an RssAggregator in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide RssAggregator</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show RssAggregator</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:  style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

- When completed, click **insert**.

A graphic defines where the RSS Aggregator appears on your template.

```

13 <body MS_POSITIONING="GridLayout">
14 <form id="Form1" method="post" runat="server">
15 <CMS:RssAggregator id="RssAggregator1" URL="http://msdn.microsoft.com/rss.xml"
16 DisplayXslt="ecmTeaser" runat="server"></CMS:RssAggregator>
17 </form>
18 </body>

```



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<CMS:RssAggregator id="RssAggregator1" URL="http://example.com/rss.xml" DisplayXslt="ecmTeaser" runat="server"></CMS:RssAggregator>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Old Search

For a description of the Old Search functions, see ["Search Server Control" on page 318](#)

This tag appears on the following menu(s):

- Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**  
or
- Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert a Old Search function:

- Move the blinking cursor to the area that you want to place it.
- From the list of Ektron CMS400.NET Content functions, click on **Old Search**

## 3. The Insert Ektron Search box is displayed.

**Insert Ektron CMS400 Search**

Search ----- Control ID:  Recursive: ☒ AJAX ☒

FolderID:

-----Select the or fill it in above-----

Order By:   Ascending

Display Type:

Max Number of results:  (0- is unlimited)

ShowSearchBoxAlways: ☒ Display Horizontal: ☐ LinkTarget:

RemoveTeaserHtml ☒ Hide: ☐ Visible: ☒ Wrap tag:

ResultTagId  PostResultsAt

ButtonImgSrc:  ButtonText:

(Positioning info - GridLayout) Style:  leave blank if none

EmptyResultMsg:  MaxCharacters:  TextBoxSize:  (If 0 ignore)

EnableAdvancedLink ☒ EnableArchived ☐ EnableArchivedBtn ☐

EnableLibHyperlinks ☒ EnableLibHyperlinksBtn ☒ EnableLibFiles ☒

EnableLibFilesBtn ☒ EnableLibImages ☒ EnableLibImagesBtn ☒

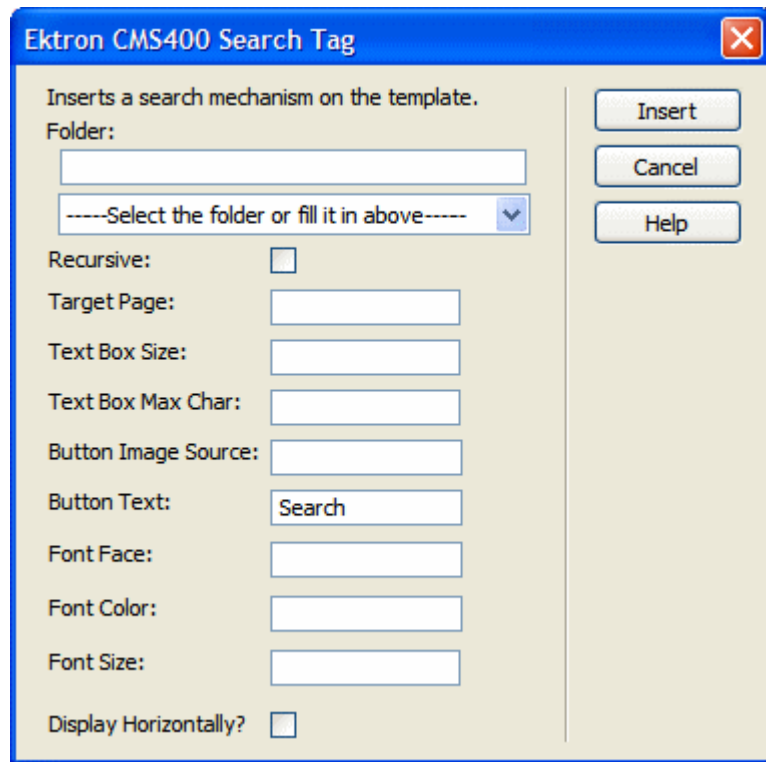
EnableDMSAssets ☒ EnableDMSAssetsBtn ☒ EnableForms ☒

EnableFormsBtn ☒ EnableContent ☒ EnableContentBtn ☒

EnableContentFieldset ☐ EnableFieldset ☐ ShowSearchOptions ☐

TargetList  ShowExtendedSearch ☒

**IMPORTANT!** When inserting a list summary using ASP, CF, JSP or PHP, a slightly different dialog box appears.



4. Use the following table to assist you with inserting a search.

Field	Description
Ajax	<p>Check this box to enable Ajax searches. When enabled, the <code>MaxResults</code> property determines the maximum number of results per page, and the <code>ResultTagId</code> property becomes active.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Enable Ajax Search</p> <p><input type="checkbox"/> <b>Unchecked</b> = Original HTML Search</p> <p>See Also: <a href="#">"Ajax Search" on page 332</a></p>
ButtonImgSrc (ASP.NET) Button Image Source (ASP, CF, JSP or PHP)	<p>If you wish to use an image for the search button, insert the location of the image in this field. If no image is specified, a standard submit button is used.</p> <p>See Also: <a href="#">"Button Image Source Examples" on page 334</a></p>
ButtonText	<p>The button text is the text that is displayed on the standard submit button, or the alt text of a search button image.</p> <p>See Also: <a href="#">"Search Button Text Examples" on page 335</a></p>

Field	Description
Control ID <hr/> <i>Used in ASP.NET only.</i> <hr/>	A unique name for the function you are inserting.
Display Horizontally	The search can be displayed either vertically or horizontally on your Web page. Check off this box if you want it to appear horizontally. See Also: <a href="#">"Display Examples" on page 335</a>
Display Type <hr/> <i>Used in ASP.NET only.</i> <hr/>	Determines how the information on the page is displayed <b>None</b> -databind only <b>ecmNavigation</b> - lists the title of every content block in the folder <b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a> and <a href="#">"Example of ecmTeaser Display" on page 102</a>
EmptyResultMsg <hr/> <i>Used in ASP.NET only.</i> <hr/>	The return message displayed, if the search returns no hits. The default is: Search returned zero results
EnableAdvancedLink <hr/> <i>Used in ASP.NET only.</i> <hr/>	Check this box to display the link that makes the advanced properties available. <input checked="" type="checkbox"/> <b>Checked</b> = Display Advanced link <input type="checkbox"/> <b>Unchecked</b> = Hide Advanced link See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a>
EnableArchived <hr/> <i>Used in ASP.NET only.</i> <hr/>	Enables users to search archived content. The archived content must be archived as Archived Remain On Site. The default is Unchecked. <input checked="" type="checkbox"/> <b>Checked</b> = Searching archived content is enabled. <input type="checkbox"/> <b>Unchecked</b> = Searching archived content is disabled, unless Include Archived is checked in the Advanced search. See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a>

Field	Description
EnableArchivedBtn <hr/> Used in ASP.NET only.	<p>Provides a check box so users can decide if they want to perform an archive search. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = The Include Archive check box is enabled in the advanced search properties. If used with EnableArchived, the check box is checked by default.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The Include Archive check box is disabled in the advanced search properties.</p>
EnableContent <hr/> Used in ASP.NET only.	<p>Enables users to search content. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Searching content is enabled.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Searching content is disabled, unless Content is checked in the Advanced search.</p> <p>See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a></p>
EnableContentBtn <hr/> Used in ASP.NET only.	<p>Provides a check box so users can decide if they want to perform a content search. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = The Content check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The Content check box is disabled in the advanced search properties.</p>
EnableContentFieldset <hr/> Used in ASP.NET only.	<p>Creates a border around the group of content items in an advanced search. These content items include Content, Forms and DMS Assets.</p> <p>The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show border around content items.</p> <p><input type="checkbox"/> <b>Unchecked</b> = No border around content items.</p> <p>See Also: <a href="#">"Fieldset Properties" on page 345</a></p>
EnableDMSAssets <hr/> Used in ASP.NET only.	<p>Enables users to search DMS Assets. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Searching DMS Assets is enabled.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Searching DMS Assets is disabled, unless DMS Assets is checked in the Advanced search.</p> <p>See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a></p>

Field	Description
EnableDMSAssetsBtn <hr/> Used in ASP.NET only.	<p>Provides a check box so users can decide if they want to perform a DMS Asset search. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = The DMS Asset check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The DMS Asset check box is disabled in the advanced search properties.</p>
EnableForms <hr/> Used in ASP.NET only.	<p>Enables users to search Forms. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Searching Forms is enabled.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Searching Forms is disabled, unless Forms is checked in the Advanced search.</p> <p>See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a></p>
EnableFormsBtn <hr/> Used in ASP.NET only.	<p>Provides a check box so users can decide if they want to perform a Forms search. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = The Forms check box is enabled in the advanced search properties. If used with EnableContent, the check box is checked by default.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The Forms check box is disabled in the advanced search properties.</p>
EnableLibFiles <hr/> Used in ASP.NET only.	<p>Enables users to search Library Files. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Searching Library Files is enabled.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Searching Library Files is disabled, unless Files is checked in the Advanced search.</p> <p>See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a></p>
EnableLibFilesBtn <hr/> Used in ASP.NET only.	<p>Provides a check box so users can decide if they want to perform a Library Files search. The default is Unchecked.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = The Files check box is enabled in the advanced search properties. If used with EnableLibFiles, the check box is checked by default.</p> <p><input type="checkbox"/> <b>Unchecked</b> = The Files check box is disabled in the advanced search properties.</p>



Field	Description
EnableLibHyperlinks <hr/> Used in ASP.NET only. <hr/>	Enables users to search Library Hyperlinks. The default is Unchecked.  <input checked="" type="checkbox"/> <b>Checked</b> = Searching Library Hyperlinks is enabled.  <input type="checkbox"/> <b>Unchecked</b> = Searching Library Hyperlinks is disabled, unless Hyperlinks is checked in the Advanced search. See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a>
EnableLibHyperlinksBtn <hr/> Used in ASP.NET only. <hr/>	Provides a check box so users can decide if they want to perform a Library Hyperlinks search. The default is Unchecked.  <input checked="" type="checkbox"/> <b>Checked</b> = The Hyperlinks check box is enabled in the advanced search properties. If used with EnableLibHyperlinks, the check box is checked by default.  <input type="checkbox"/> <b>Unchecked</b> = The Hyperlinks check box is disabled in the advanced search properties.
EnableLibImages <hr/> Used in ASP.NET only. <hr/>	Enables users to search Library Images. The default is Unchecked.  <input checked="" type="checkbox"/> <b>Checked</b> = Searching Library Images is enabled.  <input type="checkbox"/> <b>Unchecked</b> = Searching Library Images is disabled, unless Images is checked in the Advanced search. See Also: <a href="#">"Using the Advanced Search Properties" on page 339</a>
EnableLibImageBtn <hr/> Used in ASP.NET only. <hr/>	Provides a check box so users can decide if they want to perform a Library Images search. The default is Unchecked.  <input checked="" type="checkbox"/> <b>Checked</b> = The Images check box is enabled in the advanced search properties. If used with EnableLibImages, the check box is checked by default.  <input type="checkbox"/> <b>Unchecked</b> = The Images check box is disabled in the advanced search properties.
Folder	Type in the folder path to be searched. If you do not know the folder path, you can navigate to it using the Select folder drop down box.
Font Color <hr/> Used in ASP, CF, JSP and PHP only. <hr/>	The color of the font used in the search function.

Field	Description
Font Face <hr/> Used in ASP, CF, JSP and PHP only. <hr/>	The type of font to use for search function.
Font Size <hr/> Used in ASP, CF, JSP and PHP only. <hr/>	The size of the font used in the search function.
Hide <hr/> Used in ASP.NET only. <hr/>	Used to hide a search in design time and run time. <input checked="" type="checkbox"/> <b>Checked</b> = Hide search <input type="checkbox"/> <b>Unchecked</b> = Show search <hr/> <a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a> <hr/>
LinkTarget <hr/> Used in ASP.NET only. <hr/>	Defines the way a link acts when a link is clicked. The choices are: <b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named. <b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target. <b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent. <b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.
MaxCharacters (ASP.NET) Text Box Max Char (ASP, CF, JSP or PHP)	Specify the maximum amount of characters that is accepted in the text box.
Max Number of results <hr/> Used in ASP.NET only. <hr/>	Enter a number to specify how many results are displayed on the page (0=unlimited). When the Ajax property is set to checked, the way this property works changes from the maximum amount of results to the maximum results per page. This takes advantage of Ajax's ability to have paging in the searches. If zero is entered, all results appear on the same page.

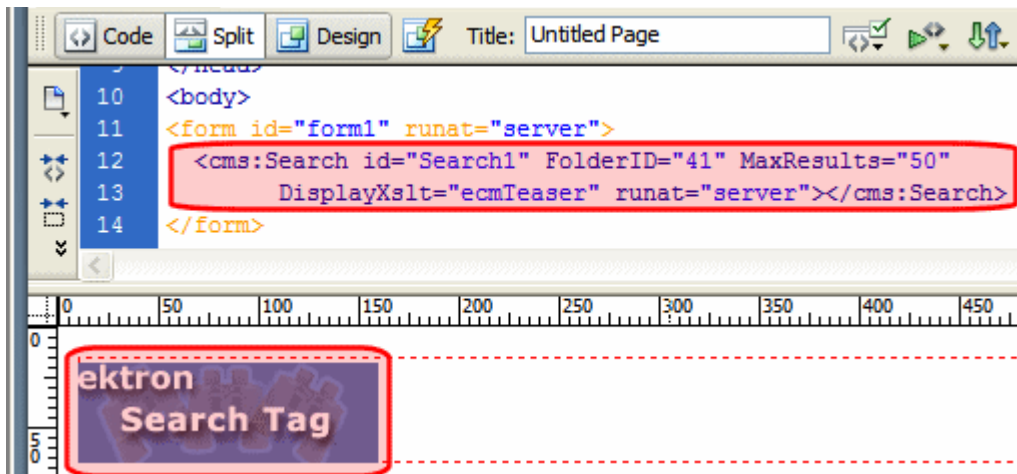
Field	Description
Order By <hr/> Used in ASP.NET only. <hr/>	<p>Specify how you want to order the search results that are generated when a search is performed on your Web site. Also specify if you want the results in ascending or descending order.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>Last Author Last Name</b> - The last author's last name</li> <li>• <b>Last Author First Name</b> - The last author's first name</li> <li>• <b>Ascending</b> - Alphabetic or numerical order</li> <li>• <b>Descending</b> - Reverse alphabetic or numerical order</li> </ul>
PostResultsAt (ASP.NET) Target Page (ASP, CF, JSP or PHP)	<p>The path of the Web page where the search results will be posted. The # symbol appears in the text box by default. The # symbol tells the control to post back to the Web page that created the search.</p> <hr/> <p><b>Important:</b> A search control must exist on the posting page.</p> <hr/>
Recursive	<p>If checked, search information is gathered from the folder selected and all of its child folders.</p>
RemoveTeaserHtml <hr/> Used in ASP.NET only. <hr/>	<p>An optional search result formatting parameter that removes all the HTML from each content item's teaser/summary before sending it to the browser.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Remove HTML</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not remove HTML</p>
ResultTagId <hr/> Used in ASP.NET only. <hr/>	<p>Specify the div tag control ID to display the search results. This property is only active when the EnableAjax property is checked.</p> <p>See Also: "<a href="#">ResultTagId</a>" on page 332</p>
Select Folder	<p>Select a folder path to be searched. If you insert the folder path, you do not need to select a folder.</p>

Field	Description
ShowExtendedSearch	<p>Allows a user to search via the metadata type: Searchable Property.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show extended search.</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide extended search.</p> <p>See Also: <a href="#">"ShowExtendedSearch" on page 336</a></p> <p>For more information on Metadata and Searchable Property, see the Administrator Manual section "Metadata".</p>
ShowSearchBoxAlways <hr/> <i>Used in ASP.NET only.</i> <hr/>	<p>If <b>unchecked</b>, the search box does not appear onPostBack.</p>
ShowSearchOptions <hr/> <i>Used in ASP.NET only.</i> <hr/>	<p>If this box is checked, the following choices appear along with the search box. This type of search is also known as a Standard search. The default setting is Checked.</p> <ul style="list-style-type: none"> <li>• All the words</li> <li>• Any of the words</li> <li>• Exact phrase</li> <li>• Match Partial words</li> </ul> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show search options</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide search options</p> <p>See Also: <a href="#">"Basic and Standard Searches" on page 340</a></p>
Style (Positioning Info - GridLayout) <hr/> <i>Used in ASP.NET only.</i> <hr/>	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
TextBoxSize	<p>Specify the size of the text box that is displayed on the Web page for users to enter the search keywords. The width is based on the number of characters.</p> <p>See Also: <a href="#">"Text Box Examples" on page 333</a></p>

Field	Description
Visible <hr/> Used in ASP.NET only.	Used to show or hide the function at run time.  <input checked="" type="checkbox"/> <b>Checked</b> = Show function <input type="checkbox"/> <b>Unchecked</b> = Hide function  <u>Note: If Hide and Visible are both checked, the property is hidden.</u>
Wrap Tag <hr/> Used in ASP.NET only.	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper <hr/> Used in ASP.NET only.	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> .  <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

- When completed, click **insert**.

A graphic defines where the search appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Search id="Search1" FolderID="41" MaxResults="50" DisplayXslt="ecmTeaser" runat="server"&gt;&lt;/cms:Search&gt;</code>
ASP	<code>&lt;% ecmSearch "\Products",1,"SearchDisplay.asp",10,25,"","Search", "Verdana","Blue","2",0,"" %&gt;</code>
ColdFusion	<code>&lt;CFMODULE template="#request.ecm.AppPath#ecmSearch.cfm" StartingFolder="\Products" Recursive="1" TargetPage="SearchDisplay.cfm" TextBoxSize="10" MaxCharacters="25" ButtonImageSrc="" ButtonText="Search" FontFace="Verdana" FontColor="Blue" FontSize="2" Horizontal="0"&gt;</code>
JSP	<code>&lt;%= ecmSearch("\Products",1,"SearchDisplay.jsp",10,25,"","Search", "Verdana","Blue","2",0,"") %&gt;</code>
PHP	<code>&lt;?php ecmSearch("\Products",1,"SearchDisplay.php",10,25,"","Search", "Verdana","Blue","","0",""); ?&gt;</code>

## Search Display

The search display custom tag creates the area where search results appear after a search is executed.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a search display function:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Insert Search Display Tag...**
3. The Insert Ektron Search Display Tag box is displayed.

Ektron CMS400 Search Display Tag

Creates area where search results will be displayed.

Maximum Number of Documents:  (0=Unlimited)

Style Info:

Show Last Modified Date: ☐

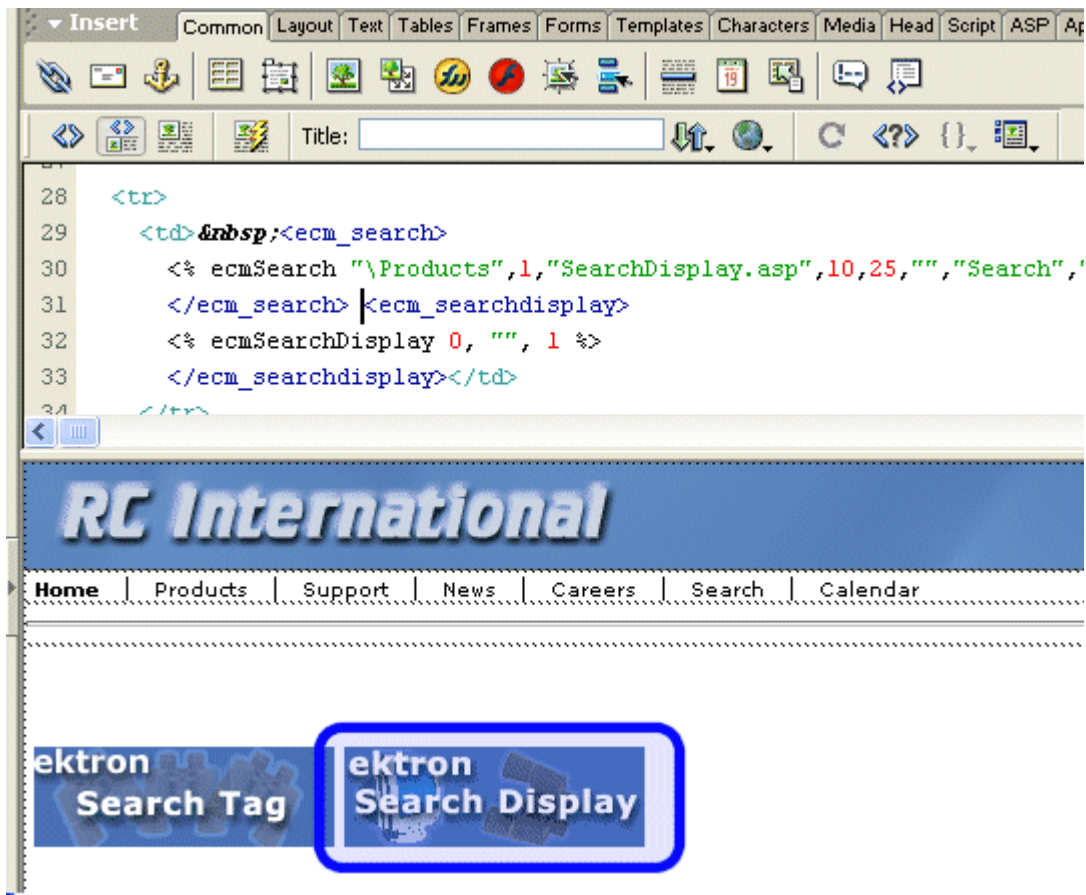
Insert Cancel Help

4. Use the following table to assist you with inserting a search display function.

Field	Description
Maximum Number of Documents	Specify the maximum amount of results that will be returned to the user when a search is performed.
Style Info	Insert an inline style sheet that will be used to format all the hyperlinks that are returned.
Show Last Modified Date	Check off this box to show the last modified date with the search result. This allows users to distinguish between updated and old content.

5. When completed, click **insert**.

A graphic to define where the search display tag has been placed on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View (with no default content block set)
.NET	In ASP.NET, you do not need to add a separate search display.
ASP	<code>&lt;% ecmSearchDisplay 0, "", 1 %&gt;</code>
ColdFusion	<code>&lt;cfmodule template="#request.ecm.AppPath#ecmSearchDisplay.cfm" MaxNumber="0" StyleInfo="" ShowDate="1"&gt;</code>
JSP	<code>&lt;%= ecmSearchDisplay(0, "", 1)%&gt;</code>
PHP	<code>&lt;?php ecmSearchDisplay(0, "", 1)?&gt;</code>

## Single Summary

For a description of the Single Summary function, see ["Single Summary Function" on page 648](#)

This tag appears on the following menu(s):

- **Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...**

To insert a Single summary:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Content functions, click on **Single Summary Tag...**
3. The Insert Ektron Single Summary Tag box is displayed.

Ektron CMS400 Single Summary Tag

Used to display a content block summary.

Content ID:

-----Select the category to retrieve list of content-----

-----Select the content block to insert into the page-----

Show Summary: ☒

Style Info:

Show Info:

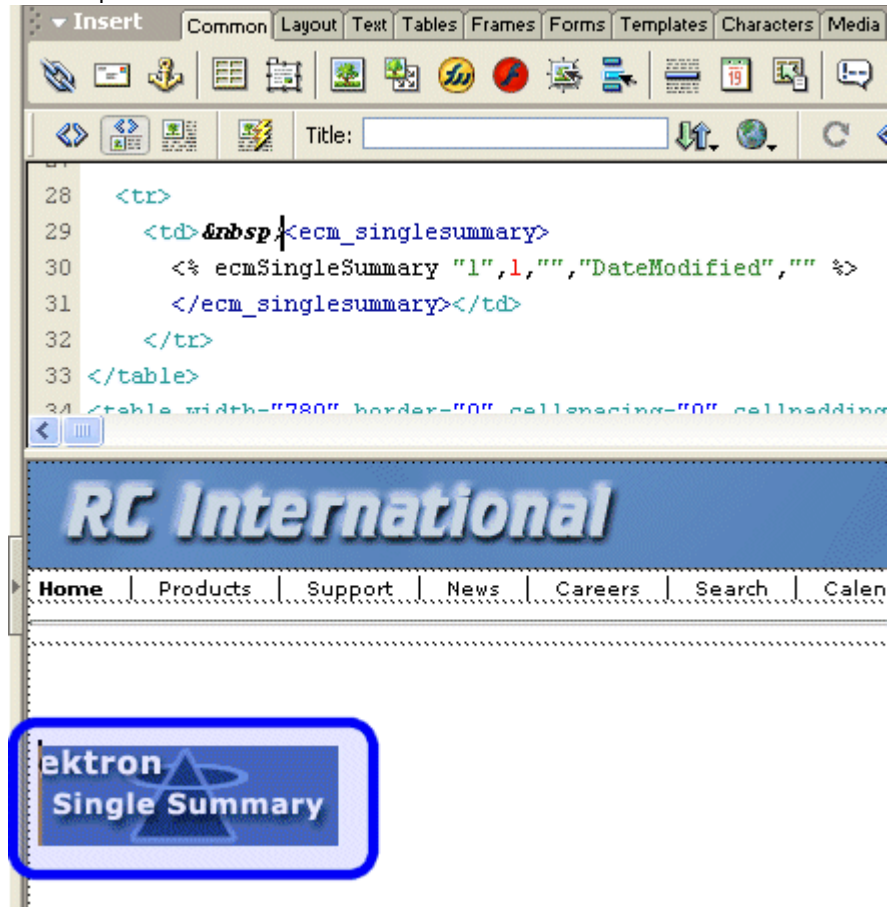


4. Use the following table to assist you with inserting a single summary tag.

Field	Description
Content ID	<ul style="list-style-type: none"><li>• Working Offline - Enter a valid content ID into the Content ID textbox.</li><li>• Working Online - The top drop-down list will be filled with a list of content folders that you have privileges to. After choosing a content folder, the bottom drop-down box will be filled with a list of all the content block titles within that category. When a title is selected, the content ID textbox will be filled with that content block's id number.</li></ul>
Show Summary	If checked, content summaries will be displayed.
Style Info	Insert any inline style information, which will be applied to the generated text hyperlink, here.
Show Info	Select the type of information that will be displayed next to the generated hyperlinks.

- When completed, click **insert**.

A graphic to define where the single summary tag will appear on your template is inserted.



Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
ASP	<% ecmSingleSummary "1",1,"","DateModified","" %>
ColdFusion	<cfmodule template="#request.ecm.AppPath#ecmSingleSummary.cfm" ContentID="1" ShowSummary="1" StyleInfo="" ShowInfo="DateModified">
JSP	<%= ecmSingleSummary("1",1,"","DateModified","") %>
PHP	<?php ecmSingleSummary("1",1,"","DateModified","") ?>

## Site Map

For a description of the Site Map function, see ["SiteMap Server Control" on page 368](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS Menu Controls...**

To insert a Site Map:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Site Map**.
3. The Site Map dialog box appears.

**Ektron CMS400 Site Map**

Site Map Control - ID:

FolderID  - Where to start Site Map from

-----Select the folder or fill it in above-----

Display Style:   DisplayXslt:

StartingLevel  MaxLevel

FlatTopLevel ☐ Hide: ☐ Visible: ☒

# of seconds - CacheInterval:  ClassName

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:   SuppressWrapperTags ☐

Insert Cancel Help

4. Use the following table to assist you with inserting a SiteMap.

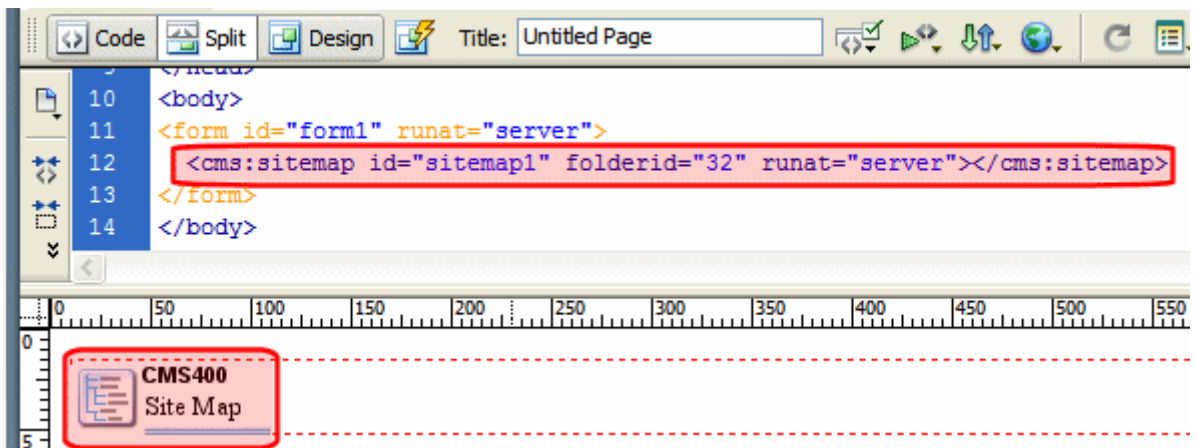
Field	Description
Control ID	A unique name for the function you are inserting.
FolderID	The numeric id of the folder that you want to use. If you do not know the folder ID, you can navigate to it using the Select folder drop down box.
DisplayStyle	Indicate how to display the site map: horizontally or vertically. The default is Horizontal.

Field	Description
DisplayXslt	<p>Determines how the information on the page is displayed.</p> <p><b>None</b> - databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the collection</p> <p>See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the collection plus the content summary</p> <p>See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p>
StartLevel	Set to the starting level of the site map. If set to 0 (zero), starts from the root.
MaxLevel	Set the maximum amount of levels to show. Set it to 0 (zero) for unlimited.
FlatTopLevel	<p>When this box is checked, include site nodes from the parent folder but not their items. Unchecking this box includes all nodes and items.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = include site nodes from the parent folder but not their items</p> <p><input type="checkbox"/> <b>Unchecked</b> = includes all nodes and items</p>
Hide	<p>Used to hide a SiteMap in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide the SiteMap</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show the SiteMap</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
# of seconds - CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a>.</p>

Field	Description
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

5. When completed, click **Insert**.

A graphic defines where the Site Map appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:sitemap id="sitemap1" folderid="32" runat="server"></cms:sitemap>

Platform	Code View
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Smart Menu

For a description of the Smart Menu function, see ["Smart Menu Server Control" on page 226](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > CMS Menu Controls...**

To insert a Smart Menu:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET Functions, click **Smart Menu**

The Smart Menu Control box appears.

**Insert Ektron CMS400 SmartMenu Tag**

**SmartMenu**

ID: SmartMenu1

DefaultMenuID:

-----Select the from a list of menus-----

DynamicParameter: None

Stylesheet: cms400demo\_menu.css

LaunchLinksInNewWindow: ☐ CacheInterval:

Enable508NavigationHeadings: ☐ Enable508Compliance: ☐

StartCollapsed: ☒ AutoCollapseBranches: ☐

StartWithRootCollapsed: ☐ ShowRootFolder: ☐

Hide: ☐ Visible: ☒

(Positioning info - GridLayout) Style: leave blank if none

Wrap tag: Span : SuppressWrapperTags: ☐

Buttons: Insert, Cancel, Help

3. Use the following table to assist you with inserting a Smart Menu.

Field	Description
Control ID	A unique name for the function you are inserting.
DefaultMenuID	The numeric id of the menu that you want to insert. If you do not know the DefaultMenuID, you can navigate to it using the Select Menu drop down box.
Select Menu	Select a menu to be used. When a menu is selected, the DefaultMenuID textbox is filled with that menu's id number. If you insert the DefaultMenuID, you do not need to select a menu.
DynamicParameter	Checks the QueryString for the menu value and replaces the menu with content when ID is specified. Leave blank to always display the menu. <b>None</b> - Use Default <b>menu_ID</b> - The control uses the menu passed as a URL parameter.

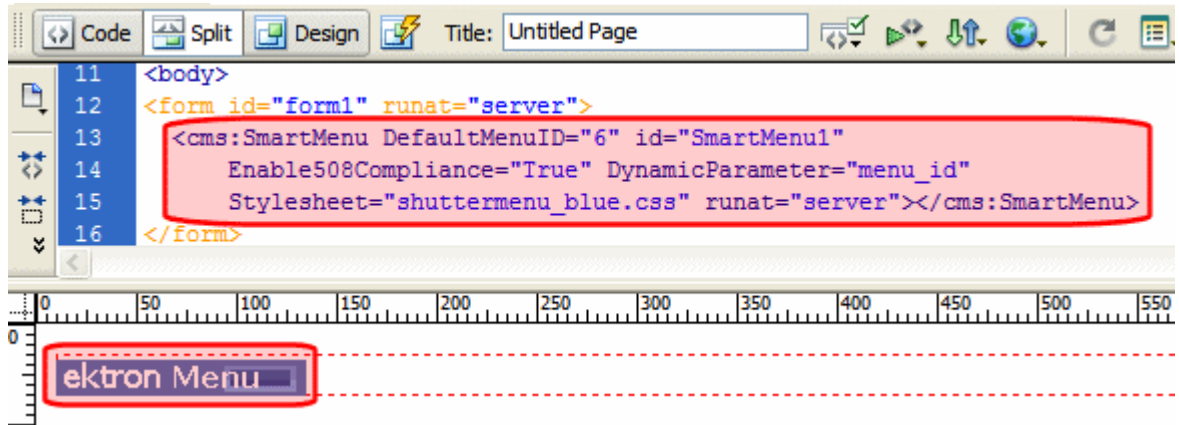
Field	Description
Stylesheet	Enter the style sheet that will determine the appearance of the menus. Menu style sheets reside in the <i>site root/workarea/csslib/smartmenu</i> folder. You can modify any of the standard style sheets or create your own.
LaunchLinksinNew Window	<p>Check this box and when a site visitor selects a menu option, it appears in a new browser window.</p> <p>If unchecked, the new page replaces the current page in the browser.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = when a site visitor selects a menu option, it appears in a new browser window</p> <p><input type="checkbox"/> <b>Unchecked</b> = the new page replaces the current page in the browser</p>
CacheInterval	Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a>
Enable508compliance	<p>If you check this box, the menu will comply with Section 508. See Also: <a href="#">"Section 508 Compliance Features of Smart Menus" on page 230</a></p> <p><input checked="" type="checkbox"/> <b>Checked</b> = menu will comply with Section 508</p> <p><input type="checkbox"/> <b>Unchecked</b> = menu does not comply with Section 508</p>
StartCollapsed	<p>If you check this box, all submenus are closed when the menu initially displays.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = all submenus are closed when the menu initially displays</p> <p><input type="checkbox"/> <b>Unchecked</b> = initially display submenus</p>
AutoCollapseBranches	<p>If you check this box, whenever a new submenu opens, all other submenus close. If unchecked, other submenus remain open when a new one opens.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = whenever a new submenu opens, all other submenus close</p> <p><input type="checkbox"/> <b>Unchecked</b> = submenus remain open when a new one opens</p>
Start With RootCollapsed	<p>If you check this box, the menu initially displays with its root menu closed.</p> <hr/> <p><a href="#">Note: This property is ignored if the <code>ShowRootFolder</code> property is unchecked.</a></p> <hr/> <p><input checked="" type="checkbox"/> <b>Checked</b> = the menu initially displays with its root menu closed</p> <p><input type="checkbox"/> <b>Unchecked</b> = the menu initially displays with its root menu open</p>



Field	Description
ShowRootFolder	<p>If you check this box, a menu option listing the title of the root folder appears. If unchecked, the title of the root folder does not appear.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = a menu option listing the title of the root folder appears</p> <p><input type="checkbox"/> <b>Unchecked</b> = the title of the root folder does not appear</p>
Hide	<p>Used to hide a Smart Menu in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Smart Menu</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Smart Menu</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

- When completed, click **insert**.

A graphic defines where the Smart Menu appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<pre> &lt;cms:SmartMenu DefaultMenuID="6" id="SmartMenu1" Enable508Compliance="True" DynamicParameter="menu_id" Stylesheet="shuttermenu_blue.css" runat="server"&gt;&lt;/ cms:SmartMenu&gt; </pre>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Standard Menu

For a description of the Standard Menu function, see ["Menu Server Control" on page 246](#).

This tag appears on the following menu(s):

- Commands > Ektron CMS400.NET (Server Controls) > CMS Menu Controls...**

To insert a Standard Menu:

- Move the blinking cursor to the area that you want to place it.
- From the list of Ektron CMS400.NET Functions, click **Standard Menu**

The Standard Menu Control box appears.

Ektron CMS400 Menu Control

Menu

Control ID:

DefaultMenuID:

-----Select the from a list of menus-----

MenuType:

Dynamic menu Parameter:

# of seconds - CacheInterval:

Hide: ☐

Visible: ☒

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:  SuppressWrapperTags ☐

Insert  
Cancel  
Help

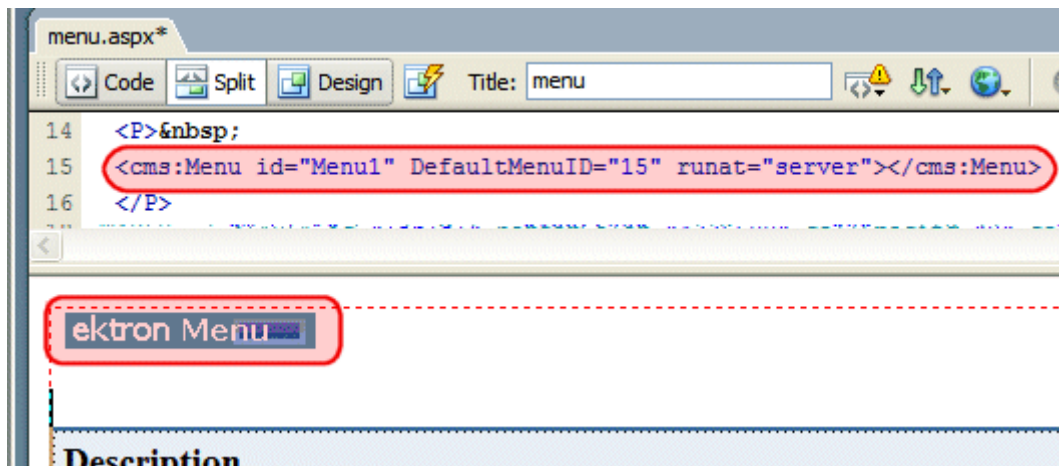
3. Use the following table to assist you with inserting a Standard Menu.

Field	Description
Control ID	A unique name for the function you are inserting.
DefaultMenuID	The numeric id of the menu that you want to insert. If you do not know the DefaultMenuID, you can navigate to it using the Select Menu drop down box.
Select Menu	Select a menu to be used. When a menu is selected, the DefaultMenuID textbox is filled with that menu's id number. If you insert the DefaultMenuID, you do not need to select a menu.
MenuType	<p>The type of menu to render.</p> <p><b>None</b> - databind only</p> <p><b>SampleMenu</b> - A sample display, formatted as a bulleted menu list</p> <p><b>TreeMenu</b> - A sample display, formatted as a folder tree. You can expand the tree by clicking on the folder icon.</p> <p>For more information on using the SampleMenu and TreeMenu Display, see <a href="#">"Using DisplayXslt Samples" on page 248.</a></p>

Field	Description
Dynamic Menu Parameter	<p>Checks the QueryString for the menu value and replaces the menu with content when ID is specified. Leave blank to always display the menu.</p> <p><b>None</b> - Use Default</p> <p><b>menu_ID</b> - The control uses the menu passed as a URL parameter.</p>
CacheInterval	<p>Sets the amount of time the server control's data is cached. The default is 0 (zero). This is the amount of time, in seconds, a control's data is cached. For example, if you want to cache the data for five minutes, you set the <code>CacheInterval</code> property to 300 (three hundred). See Also: <a href="#">"Caching with Server Controls" on page 42</a></p>
Hide	<p>Used to hide a menu in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><a href="#">Note: If Hide and Visible are both checked, the property is hidden.</a></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page.</p> <p>For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The <code>&lt;span&gt;</code> tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The <code>&lt;div&gt;</code> tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>

4. When completed, click **insert**.

A graphic defines where the Standard Menu appears on your template.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<code>&lt;cms:Menu id="Menu1" DefaultMenuID="15" runat="server"&gt;&lt;/cms:Menu&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## Web Search

For a description of the Web Search Function, see ["IndexSearch Server Control" on page 351](#).

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert an Web Search:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **Web Search**.

3. The Insert Web Search box is displayed.

**Ektron Web Search**

**WebSearch**

Control ID:

FolderId:  Recursive ☒

-----Select the folder or fill it in above-----

ButtonImgSrc  ButtonText

DisplayXslt  ecmNavigation

PostResultsAt  ResultTagId

Stylesheet  TemplateMarkup

TextBoxSize  ShowSearchBoxAlways ☒

ShowSearchOptions ☒ EnableAjax ☒ EnableAdvancedLink ☒

EnableCallback ☒ MaxCharacters

OrderBy  Title OrderDirection  Ascending

LinkTarget  \_self

Hide ☐ Visible ☒

(Positioning info - GridLayout) Style:  leave blank if none


Wrap tag:  Span SuppressWrapperTags ☐

Insert Cancel Help

4. Use the following table to assist you with inserting an Web Search.

Field	Description
Control ID	A unique name for the function you are inserting.
FolderID	The numeric id of the folder that you want to search. If you do not know the folder path, you can navigate to it using the Select folder drop down box.
Recursive	In the <b>FolderID</b> property, you specify a folder whose content is searched on this map. To extend the search to all folders below this folder, check this box.

Field	Description
Select Folder	Select a folder path to be searched. When a folder is selected, the folderID textbox is filled with that folder's id number. If you insert the Folder ID, you do not need to select a folder.
ButtonImgSrc	If you want to display an image on the submit button, enter the server path to it.
ButtonText	The text used for the button if no image source is identified. If an image source is identified, this is alternative text for the button.
DisplayXslt	<p>Determines how the information on the page is displayed</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder</p> <p>See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary</p> <p>See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p>
PostResultsAt	<p>Enter the Web form on which the search results will post.</p> <p>To display results on the Web page from which the search was initiated, enter a pound sign (#) or leave the field blank.</p> <hr/> <p><b>Important:</b> A Web Search server control must exist on the posting page.</p> <hr/>
ResultTagId	<p>Lets you designate where search results appear. You can place search criteria in one area of a Web form and the results in another.</p> <p>For example, you have the following tag.</p> <pre>&lt;span id="results"&gt;&lt;/span&gt;</pre> <p>In this case, enter <b>results</b> for the property value.</p>
Stylesheet	<p>Specify the location of the style sheet for the search results page. Set the location relative to the site root folder. For example: <code>csslib\mytest.css</code>.</p> <p>Leave blank to use the default style sheet, <code>\webroot\Workarea\csslib\global.css</code>.</p>

Field	Description
MarkupLanguage	<p>Identify the template markup file that controls the display of the search results. For example, mysearchmarkup.ekml.</p> <p>If the *.ekml file is located in the same folder as the Web form containing the server control, just enter its name. If the file is in another folder, enter the path relative to site root. For example,  \cms400Min\workarea\customfiles\markup\mysearchmarkup.ekml.</p> <p>If you enter a valid EkML file, the Displayxslt property value is ignored.</p> <p>See Also: "Ektron Markup Language" on page 754 and "websearch.ekml" on page 798</p>
Text Box Size	The size of the <b>Search</b> text box for user input. The width is based on the number of characters.
ShowSearchBoxAlways	<p>If set to <b>unchecked</b>, the search box does not appear when the results are posted to the Web page.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = search box always appears</p> <p><input type="checkbox"/> <b>Unchecked</b> = the search box does not appear when the results are posted to the Web page</p>
ShowSearchOptions	<p>If this box is checked, the following options appear above the <b>Search</b> box.</p> <ul style="list-style-type: none"> <li>• Site</li> <li>• Html</li> <li>• Documents</li> <li>• Images</li> <li>• Multimedia</li> </ul> <p><input checked="" type="checkbox"/> <b>Checked</b> = show search options</p> <p><input type="checkbox"/> <b>Unchecked</b> = hide search options</p>
EnableAdvancedLink	<p>Set to true to display the link to the Advanced Search page (see image below).</p> <p>: Multi Media</p> 
MaxCharacters	<p>The maximum number of characters the <b>Search</b> text box accepts.</p> <p>If you enter a value less than 50, it is a good idea to set the <b>TextBoxSize</b> property to the same number.</p>

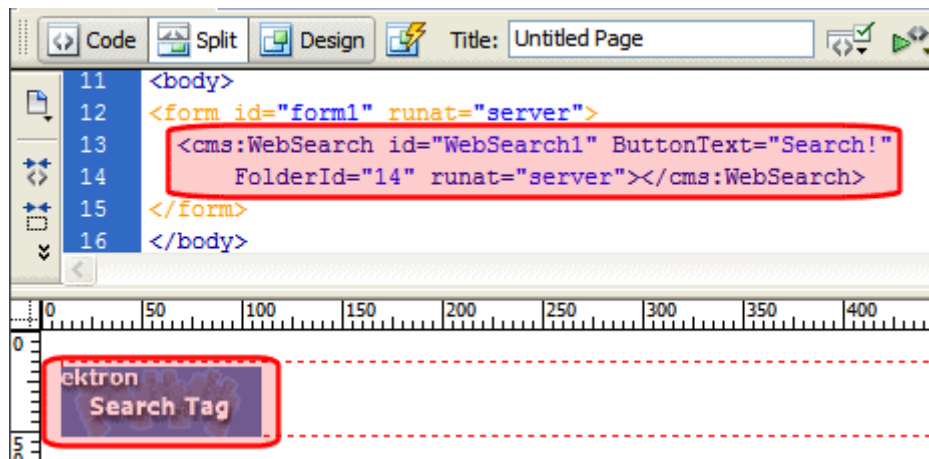


Field	Description
Order By	<p>Specify how you want to order the search results that are generated when a search is performed on your Web site.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>Last Author Last Name</b> - The last author's last name</li> <li>• <b>Last Author First Name</b> - The last author's first name</li> </ul>
OrderDirection	<p>Specify if you want the results in ascending or descending order.</p> <ul style="list-style-type: none"> <li>• <b>Ascending</b> - Alphabetic or numerical order</li> <li>• <b>Descending</b> - Reverse alphabetic or numerical order</li> </ul>
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like “_self” if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like “_self” if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>
Hide	<p>Used to hide an Web Search in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide Web Search</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show Web Search</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>

Field	Description
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page. For example: style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag	Allows a developer to specify a server control's tag. The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element. <b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.
SuppressWrapper	Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b> . <input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags <input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags

5. When completed, click **insert**.

A graphic defines where the Web Search appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<code>&lt;cms:WebSearch id="WebSearch1" ButtonText="Search!" FolderId="14" runat="server"&gt;&lt;/cms:WebSearch&gt;</code>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.

Platform	Code View
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

## XML Index Search

For a description of the XML Index Search Function, see "[IndexSearch Server Control](#)" on page 351.

This tag appears on the following menu(s):

- **Commands > Ektron CMS400.NET (Server Controls) > Content Controls...**

To insert an XML Index Search:

1. Move the blinking cursor to the area that you want to place it.
2. From the list of Ektron CMS400.NET functions, click **XML Index Search**.

3. The Insert XML Index Search box is displayed.

Used to display a list of content block from a folder.

Control ID:

XML ConfigID:

-----Select the XML Configuration-----

FolderID:

-----Select the folder or fill it in above-----

Search Parm XML:

Weighted: ☒

ShowSearchBoxAlways: ☒

Recursive: ☒ LinkTarget:

Order By:

Max Number of results:

Display Type:

Dynamic Content Parameter:

EncodeContentHtml ☒

Hide: ☐

Visible: ☒

(Positioning info - GridLayout) Style:  leave blank if none

Wrap tag:   SuppressWrapperTags ☐

Button Text:

Hits Marker:

Insert Cancel Help

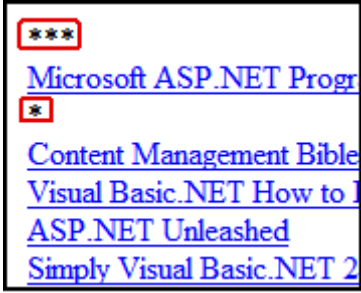
4. Use the following table to assist you with inserting an XML Index Search.

Field	Description
Control ID	A unique name for the function you are inserting.
XML ConfigID	ID of the Smart Form whose indexed fields appear on the search screen. If you do not know the XML ConfigID, you can navigate to it using the Select Smart Form drop down box.

Field	Description
Select XML Configuration	The drop-down box is populated with a list of all the XML Configuration titles. When a title is selected, the XML ConfigID textbox is filled with that XML Configuration's id number. If you insert the XML ConfigID, you do not need to select a XML Configuration.
FolderID	The numeric id of the folder that you want to search. If you do not know the folder path, you can navigate to it using the Select folder drop down box.
Select Folder	Select a folder path to be searched. When a folder is selected, the folderID textbox is filled with that folder's id number. If you insert the Folder ID, you do not need to select a folder.
Search Parm XML	This property preserves settings that a developer would create in Visual Studio.
Weighted	<p>If this value is <b>checked</b>, content blocks are ranked by the number of criteria matches.</p> <p>For example, if a content block matches all search criteria, it appears at the top of the results page. A hit marker appears for each match.</p> <p>If a second content block only matches one criterion, it also appears on the results page underneath the first content block. One asterisk appears above the page's title to indicate the single match.</p> <p><input checked="" type="checkbox"/> <b>Checked</b>- display every content block that matches <i>at least one</i> search criteria</p> <p><input type="checkbox"/> <b>Unchecked</b> - show only content blocks that match <i>all</i> search criteria</p>
ShowSearchBoxAlways	If set to <b>unchecked</b> , the search box does not appear on when the results are posted to the Web page.
Recursive	Determines whether the XML Index Search Control uses this folder's child folders in the search.
LinkTarget	<p>Defines the way a link acts when a link is clicked.</p> <p><b>_blank</b> - This target causes the link to always be loaded in a new blank window. This window is not named.</p> <p><b>_self</b> - This target causes the link to always load in the same window the anchor was clicked in. This is useful for overriding a globally assigned base target.</p> <p><b>_parent</b> - This target makes the link load in the immediate frameset parent of the document. This defaults to acting like "<b>_self</b>" if the document has no parent.</p> <p><b>_top</b> - This target makes the link load in the full body of the window. This defaults to acting like "<b>_self</b>" if the document is already at the top. It is useful for breaking out of an arbitrarily deep frame nesting.</p>

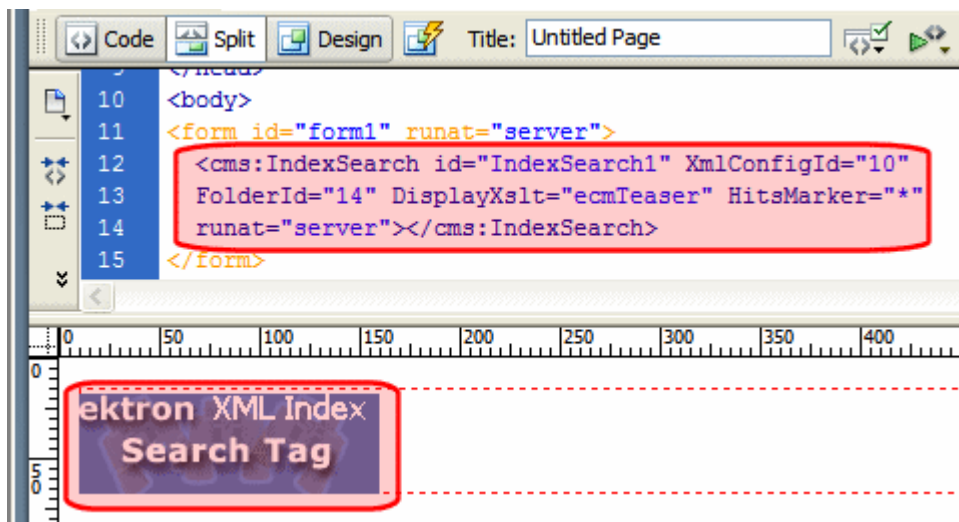
Field	Description
Order By	<p>Specify how you want to order the search results that are generated when a search is performed on you Web site. Also specify if you want the results in ascending or descending order.</p> <ul style="list-style-type: none"> <li>• <b>Title</b> - The title of the content block</li> <li>• <b>Date Created</b> - The date the content block was created</li> <li>• <b>Date Modified</b> - The date the content block was last modified</li> <li>• <b>Last Author Last Name</b> - The last author's last name</li> <li>• <b>Last Author First Name</b> - The last author's first name</li> <li>• <b>Ascending</b> - Alphabetic or numerical order</li> <li>• <b>Descending</b> - Reverse alphabetic or numerical order</li> </ul> <p>When you choose weighted search results, Ascending and Descending use how many criteria match your search. For example, if you have weighted results, and you choose Ascending, the results that match the most criteria appear first.</p>
Max Number of results	The Maximum number of content blocks returned (0=unlimited).
Display Type	<p>Determines how the information on the page is displayed</p> <p><b>None</b>-databind only</p> <p><b>ecmNavigation</b> - lists the title of every content block in the folder See Also: <a href="#">"Example of ecmNavigation Display" on page 101</a></p> <p><b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary See Also: <a href="#">"Example of ecmTeaser Display" on page 102</a></p>
Dynamic Content Parameter	<p>Checks the QueryString for the content block ID value and replaces the search with content when ID is specified. Leave blank to always display the search. By making content dynamic, when the content displayed on a Web page contains a link to another piece of content, you can display the second piece of content in the same Web page space.</p> <p><b>None</b> - Use Default</p> <p><b>ID</b> - The control uses the content block passed as a URL parameter.</p>
EncodeContentHtml	<p>When checked, Content HTML is encoded when it is returned.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Content HTML is encoded</p> <p><input type="checkbox"/> <b>Unchecked</b> = Content HTML is not encoded</p>

Field	Description
Hide	<p>Used to hide an IndexSearch in design time and run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Hide IndexSearch</p> <p><input type="checkbox"/> <b>Unchecked</b> = Show IndexSearch</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Visible	<p>Used to show or hide the function at run time.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Show function</p> <p><input type="checkbox"/> <b>Unchecked</b> = Hide function</p> <hr/> <p><i>Note: If Hide and Visible are both checked, the property is hidden.</i></p> <hr/>
Style (Positioning Info - GridLayout)	<p>Set position information of a control as absolute, when using a GridLayout Web page. For example:</p> <p>style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"</p>
Wrap Tag	<p>Allows a developer to specify a server control's tag.</p> <p>The default is <b>Span</b>.</p> <p><b>Span</b> - The &lt;span&gt; tag is used to designate an in-line portion of an HTML document as a span element.</p> <p><b>Div</b> - The &lt;div&gt; tag is used when you want to apply attributes to a block of code.</p>
SuppressWrapper	<p>Suppresses the output of the span/div tags around the control. The default is <b>Unchecked</b>.</p> <p><input checked="" type="checkbox"/> <b>Checked</b> = Suppress wrapper tags</p> <p><input type="checkbox"/> <b>Unchecked</b> = Do not suppress wrapper tags</p>
Button Text	<p>The text that appears on the button.</p>

Field	Description
Hits Marker	<p>This symbol shows how many search criteria each hit matched.</p> <p>The following example shows the first hit met three of the criteria. The second group met one criterion.</p> 

- When completed, click **insert**.

A graphic defines where the XML Index Search appears on your template.



Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<pre>&lt;cms:IndexSearch id="IndexSearch1" XmlConfigId="10" FolderId="14" DisplayXslt="ecmTeaser" HitsMarker="*" runat="server"&gt;&lt;/cms:IndexSearch&gt;</pre>
ASP	This function is not available as an Ektron Dreamweaver Extension for ASP.
ColdFusion	This function is not available as an Ektron Dreamweaver Extension for ColdFusion.



Platform	Code View
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	This function is not available as an Ektron Dreamweaver Extension for PHP.

# Standard ASP.NET Methods

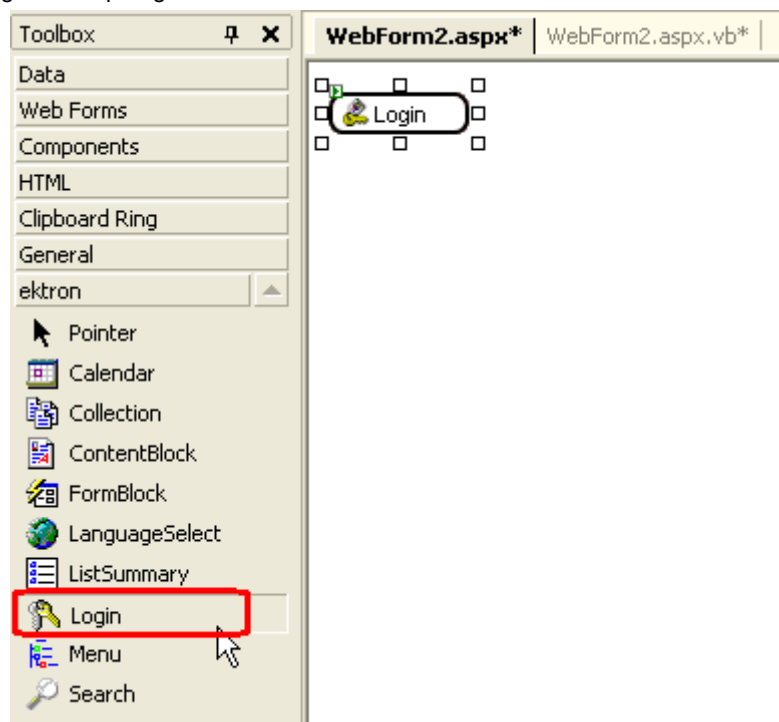
Use these methods to customize **Ektron CMS400.NET**. To use them, insert them onto your template pages to invoke content blocks, list summaries, and other objects stored in your database. They are the primary building blocks of your **Ektron CMS400.NET** Content Management System.

## Replacement of Standard Methods

Beginning with Release 4.7, **Ektron CMS400.NET** utilizes ASP.NET server controls, which replace most Standard ASP.NET methods. With **Ektron CMS400.NET**'s new toolbox of pre-built, reusable, data-bindable ASP.NET controls, Ektron empowers developers working in Visual Studio 2005 to rapidly integrate Ektron CMS components into their templates and sites. For more Information on Ektron CMS400.NET controls, see ["Introduction to Ektron CMS400.NET Server Controls" on page 7](#).

The following examples shows the Login server control added by drag and drop vs. adding a Login method by the ASP.NET method.

Drag and Drop Login server control



Adding a Login function via an ASP.NET method.

```
<tr>
 <td>
 <%=AppUI.ecmLogin()%>
```

```

 </td>
 </tr>

```

### Best Practice

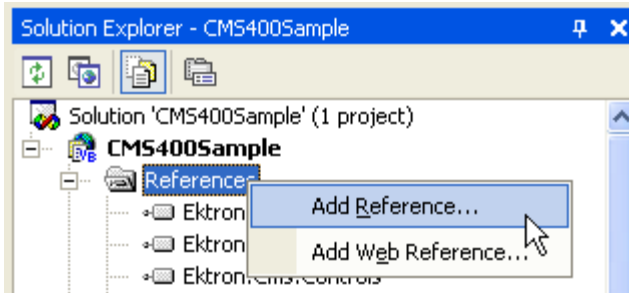
Ektron recommends using CMS400.NET server controls when available over Ektron standard ASP.NET methods. The methods are being replaced by the server controls.

You can still use the standard ASP.NET methods in **Ektron CMS400.NET**.

## Referencing Methods in Visual Basic

Before you can use methods with Visual Basic, you must add a reference to them from Visual Basic. Follow these steps to do so.

1. Open Visual Basic.
2. Choose or add a project.
3. In Solution Explorer, right click **References** and choose **Add Reference**.

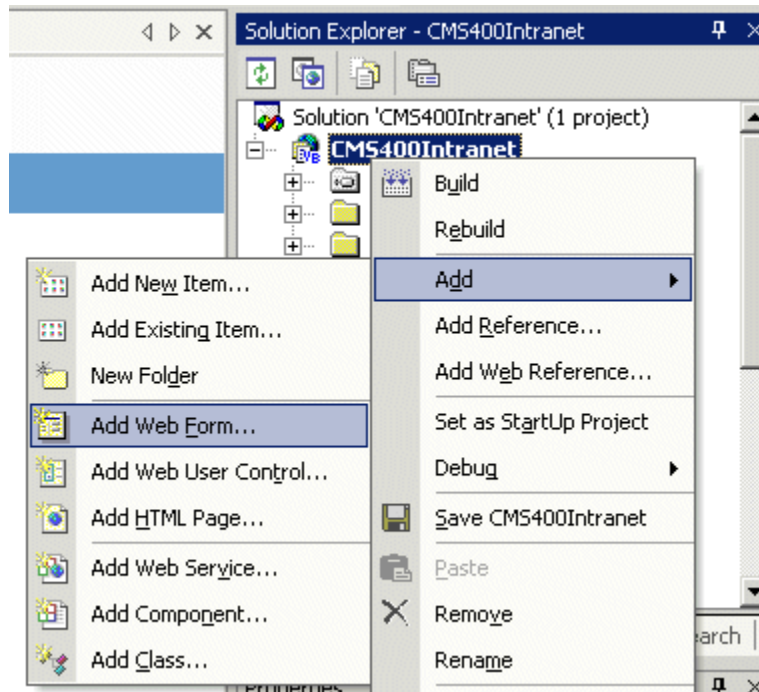


4. In the Add Reference dialog box, click **Browse**.
5. Go to *webroot/siteroot/Bin*.
6. To load the standard methods click *Ektron.Cms.UI.CommonUI.dll*. To load the XML Indexing methods, click *Ektron.Cms.Common.dll*.
7. Click **Open**, then **OK**.

## Adding a New Web Form

To add a new Web form, follow these steps.

1. Open your project in Visual Studio 2005.
2. Right click the project.
3. Click **Add > Add Web Form**.



4. Click **Web Form**.
5. In the code behind, add an Imports `Ektron.Cms.UI.CommonUI` statement.

## Requirements for Standard Methods

Ektron CMS400.NET maintains and displays content through the use of Ektron ASPX methods and ASP.NET assembly objects. To use the methods, include the following code in your template:

```
<%@ Page Language="vb" AutoEventWireup="false" %>
<%@ Import Namespace="Ektron.Cms.UI.CommonUI" %>
<% Dim AppUI As New ApplicationAPI %>
```

Each line is explained below.

Line	Description
<%@ Page Language="vb" AutoEventWireup="false" %>	A generic .net line of code that should always be placed in the template
<%@ Import Namespace="Ektron.Cms.UI.CommonUI" %>	An import statement that calls the namespace <code>Ektron.CMS.UI.CommonUI</code> to display information from the .net assembly file

Line	Description
<% Dim AppUI As New ApplicationAPI %>	Creates an instance of the ApplicationUI class that is used by each method

These lines call the Ektron .net assembly object to allow the page to function properly. They must appear at the top of each template file.

## Standard Methods

To customize your Ektron CMS400.NET application, you need to be familiar with these methods.

Method	Description	More Information
Add Content	Programmatically adds a content block(s) to Ektron CMS400.NET	"Addcontent" on page 600
Collection	Displays links generated by collections.	"Collection Server Control" on page 95
Content Block	Displays and manages a content block.	"ContentBlock Server Control" on page 108
Event Calendar	Displays an event calendar	"Calendar Server Control" on page 88
Form	Displays a content block that is associated with a form.	"FormBlock Server Control" on page 159
IndexSearch	Displays on a Web page a set of search fields. A site visitor uses the page to search for information on your Web site.	"IndexSearch Server Control" on page 351
List Summary	Displays a list of content block summaries.	"ListSummary Server Control" on page 178
Login	Displays the login or logout button, which lets the user log in and use Ektron CMS400.NET.	"Login Server Control" on page 186
Metadata	Displays the metadata in the source of the Web page displaying the content.	"MetaData Server Control" on page 261
Menu	Customizes the DHTML menus feature	"DhtmlMenu Server Control" on page 254

Method	Description	More Information
Search	Inserts a text box with a search button in the template. When the user completes the search screen, it looks through XHTML content on the Web site.	"Search Server Control" on page 318
ShowSiteLanguage	Lets a site visitor select a language in which to view site.	"LanguageSelect Server Control" on page 176

## Addcontent

The addcontent method lets you programmatically add a new content block to Ektron CMS400.NET without logging in. This method can be used for several applications, such as

- A Web form allowing you to collect data from users about the content block
- A loop function that moves information from a database to an Ektron CMS400.NET database

### IMPORTANT!

Every string that you pass to the function must be HTML encoded. Here is the VB.Net syntax for encoding a string.

```
myTitle = "John's Book"
myTitle = Server.HtmlEncode(myTitle)
```

Explained in the table below are the attributes for the add content block function.

Attribute	Description
Username	The username of the user to be authenticated for permissions to see whether the user is able to add a content block.
Password	The password for the username being authenticated.
Domain	If using Active Directory, specify the domain needed to authenticate the user.
Content_title	Specify a title for the content block being added.
Content_comment	Specify a historical comment for the content being added.
ContentHTML	Actual content for the content block.
SummaryHTML	Summary information about the content block.
ContentLanguage	Language of content.

Attribute	Description
FolderID	ID number of the content folder where the content block will be added to.
GoLive	Date and time the content will go live. <u>Must match date/time format set in the Web.config key ek_dateFormat.</u>
EndDate	Date and time the content will expire. <u>Must match date/time format set in the Web.config key ek_dateFormat.</u>
MetaInfo	<p>Specify the information for the metadata which will be used as a reference point for finding other words or information.</p> <p>An example is:</p> <pre>&lt;metadata&gt;&lt;meta id="3"&gt;Title&lt;/meta&gt;&lt;/metadata&gt;</pre> <p>Where ID is the metadata definition ID whose value is being set. The value is set between the meta tags and the meta tag is repeatable for as many metadata definitions as you want to set for the given content block.</p>
ErrString	If there is a error then errString contains the error, otherwise errString is empty.

## Add Content Example

The following is an example of the add content function in a template:

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
 Dim m_apiObj As Ektron.Cms.UI.CommonUI.ApplicationAPI
 Dim newContentId As Integer = 0
 Try
 m_apiObj = New Ektron.Cms.UI.CommonUI.ApplicationAPI
 newContentId = m_apiObj.AddContent(_
 "admin", _
 "admin", _
 "", _
 "Content Title", _
 "Comment - this content added through API!", _
 "This is the body of the content.", _
 "This is the <i>summary</i> for the content.", _
 "1033", _
 "0", _
 "06-27-2006 4:09 PM", _
 "", _
 "<metadata><meta id=""3"">ContentTitle</meta></metadata>", _
 "")
 Literal1.Text = "Content Added, ID = " + newContentId.ToString
 Catch ex As Exception
 Literal1.Text = "Error, AddContent Failed: " + ex.Message
 Finally
 m_apiObj = Nothing
 End Try
End Sub
```

The previous example has the following properties:

- The user “admin” is used to be authenticated.
- The title of the new content block is “Content Title”.
- The comment for the content is “Comment”.
- The body of the content is “This is the **body** of the content”.
- The summary of the content block is “This is the *summary* for the content”.
- The language ID = 1033, or English.
- The folder ID = 0 is where the content is added.



- The start date is 06-27-2006.
- No end date is specified.
- The metadata definition ID being set is "3" and the value is ContentTitle.

When this function is executed, the content is added to Ektron CMS400.NET. Displayed here is the record of the content in the workarea.

View Content "Content Title"

View: English (U.S.) | Add: -

Properties | Content | Summary | Metadata | Comment | Tasks | We

**Properties**

<b>Content Title:</b>	Content Title
<b>Content ID:</b>	490
<b>Content Language</b>	English (U.S.)
<b>Status:</b>	Approved (Published)
<b>Last User To Edit:</b>	Application Administrator
<b>Last Edit Date:</b>	6/28/2006 2:35:44 PM
<b>Start Date:</b>	6/27/2006 4:09:00 PM
<b>End Date:</b>	[None Specified]
<b>Action on End Date</b>	[None Specified]
<b>Date Created:</b>	6/28/2006 2:35:40 PM
<b>Approval Method:</b>	Do Not Force All Approvers
<b>Approvals:</b>	[None Specified]
<b>Smart Form Configuration:</b>	[None Specified] HTML Content Assumed
<b>Template</b>	dynamic.aspx
<b>Path</b>	\
<b>Rating</b>	this content has not been rated.
<b>Content Searchable:</b>	True

# Custom ASP Functions

Ektron CMS400.NET displays content on your Web site through the use of custom Ektron ASP functions and ComObjects. Your template you must include the following file to be able to use the custom functions:

```
<!-- #include file="applicationAPI.asp" -->
```

This file (**application.asp**) calls the file with the custom Ektron functions to allow the page to function properly. The `include` statement must appear at the top of every template file that uses an Ektron CMS400.NET custom ASP function.

## IMPORTANT!

`ContentWS.asmx` is the web services (WSDL) file that the custom ASP functions use to talk to **Ektron CMS400.NET**. The WSDLpath located in the `applicationAPI.asp` file must be set to location of the WSDL file. The default WSDLpath is `http://localhost/cms400Min/WorkArea/WebServices/ContentWS.asmx`. If you have changed the location of `ContentWS.asmx` you need to update the WSDLpath. For example, if your site is `www.example.com`, the WSDLpath would be `http://www.example.com/WorkArea/WebServices/ContentWS.asmx`.

To customize Ektron CMS400.NET, you need to be familiar with these tags.

Function	Description	More Information
<b>Calendar Functions</b>		
Calendar	Displays a calendar with different display types. For example, a daily view calendar.	"Calendar Function" on page 674
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 671
<b>Collection Functions</b>		
Collection	Displays links generated by collections	"Collection Function" on page 657
CollectionXML	Creates XML for a collection by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML Collection Function" on page 663
Random Content	Randomly displays the content of a collection item	"Random Content" on page 669

Function	Description	More Information
Random Summary	Randomly displays the summary of a collection item	"Random Summary" on page 670
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 661
<b>Content Functions</b>		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> programmatically.	"Add Content" on page 617
Content Block	Displays and manages a content block	"Content Block Function" on page 609
Form	Displays a content block associated with a form	"Form Function" on page 678
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 680
<b>List Summary Functions</b>		
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 647
List Summary	Displays a list of content block summaries	"List Summary Function" on page 635
List Summary XML	Creates XML for a list summary by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML List Summary Function" on page 642
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS Summary Function" on page 652
Single Summary	Displays the summary for a content block	"Single Summary Function" on page 648
<b>Login Functions</b>		
Login	Adds the login or logout button	"Login Function" on page 607



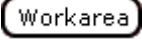
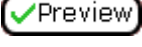

Function	Description	More Information
<b>Menu Functions</b>		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 688
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 688
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu" on page 692
<b>Metadata Functions</b>		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 681
<b>Search Functions</b>		
Search	Inserts a text box with a search button	"Search Functions" on page 621
Search Display	Creates an area to display search results	"Search Display Function" on page 631
<b>Site Language Functions</b>		
ShowSiteLanguage	Lets a site visitor select a language in which to view site.	"Show Site Language Function" on page 687

# Login Function

The login function adds a login button on the template when displayed in a browser. Here it the format of the login function.

<% ecmLogin %>

This tag is responsible for displaying the following buttons on the Web page.

Button	Name	Description
	Login	When user is not logged into Ektron CMS400.NET, this button appears. Click this button to let the user log in to the application.
	Logout	Once logged in, this button replaces the login button to allow the user to log out from the Ektron CMS400.NET Web site.
	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.
	Preview Off	Turns off preview mode.

## Login Tag Example

The following example places a login button in a table cell inside an asp template.

<tr>  
 <td>  
 <% ecmLogin %>  
 </td>  
</tr>

When inserted properly, a login button appears on the Web page.



Clicking the login button opens a login window in which a user enters the username and password. Upon successful completion, the user is logged in to the Ektron CMS400.NET Web site.

## Placing Your Login Button

You can add as many login buttons to a template as you like. You might want to include a login button on each template, or have a special Web page called `login.asp`, from which users can log in without the public being able to access the login page.

# Content Block Function

The content block tag displays a content block on an Ektron CMS400.NET Web page. There are two functions to choose from:

- `ecmContentBlock`
- `ecmContentBlockEX`

`ecmContentBlock` allows you display a content block without having to define an XSLT.

With `ecmContentBlockEX`, you can display XML content by defining an XSLT or applying one of the predefined XSLTs located in **Ektron CMS400.NET**. For more information on predefined XSLTs, see the **Administrator's Manual** > "Managing XML" > "Adding a Smart Form" > "Adding a Smart Form Using External XML Files".

Here is the format for the `ecmContentBlock` function.

```
<%ecmContentBlockEX ID %>
```

Here is the format for the `ecmContentBlockEX` function.

```
<%ecmContentBlockEX ID,
"XSLT",
Override %>
```

**NOTE** As an alternative, you can use the multipurpose function, which can display either a content block or a content block associated with a form. For more information, see ["MultiPurpose Function" on page 680](#).

The table below explains the attributes of the `ecmContentBlock` function.

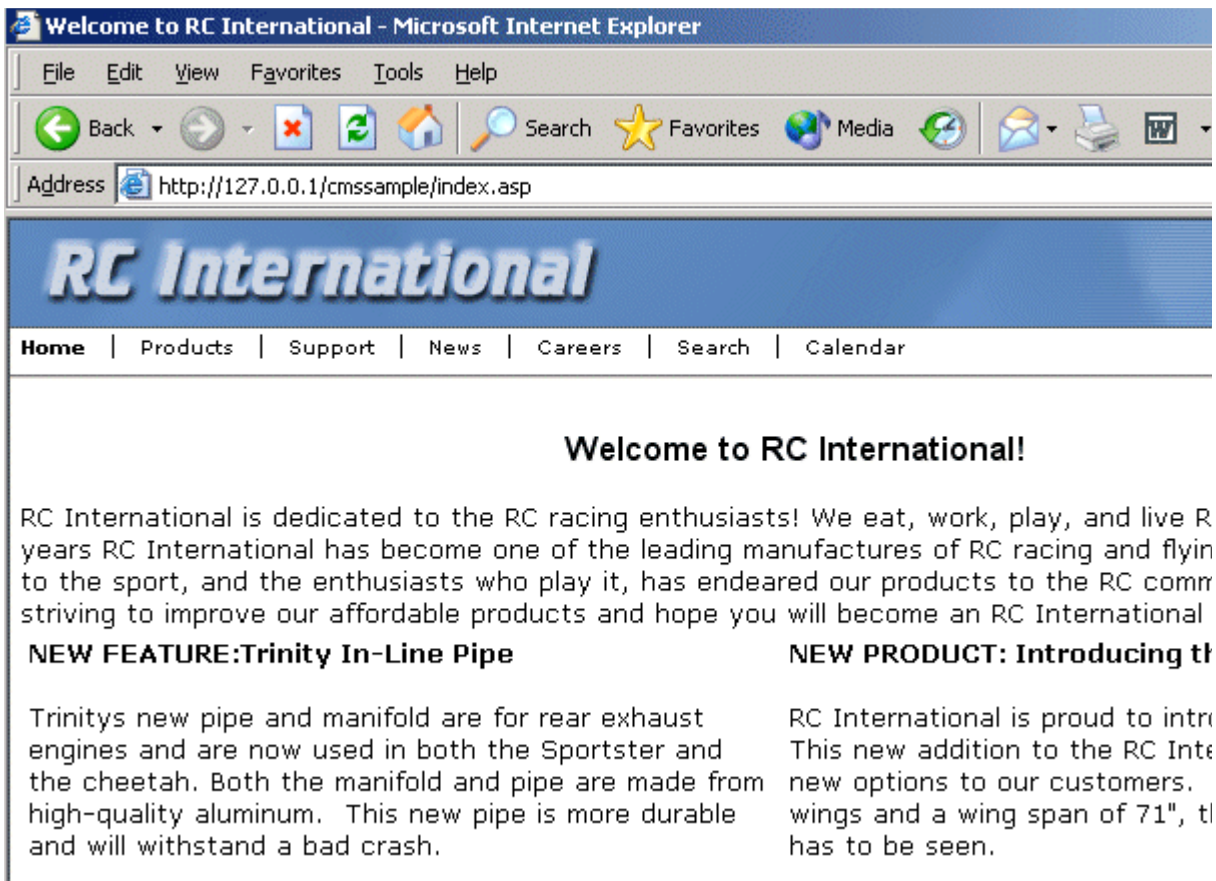
Attribute	Description	Required
ID	ID number assigned of the content block that will appear	Yes

The table below explains the attributes of the `ecmContentBlockEX` function.

Attribute	Description	Required
ID	ID number assigned of the content block that will appear	Yes
XSLT	File name and path to a valid XSLT to transform the XML content. See Also: <a href="#">"The Function's Arguments" on page 614</a>	No

Attribute	Description	Required
Override	If no XSLT is specified, then enter 1 to use XSLT1, 2 for XSLT2, or 3 for XSLT3. See Also: <a href="#">"The Function's Arguments" on page 614</a>	No

In normal view, these tags retrieve the content block from the database and displays it in the template.






If a user is logged into Ektron CMS400.NET, the content block tag displays a border around the content block and when they right click on the content, a menu is provided for managing it.

## Our Company Intranet

At the heart of Acme is our commitment to our customers, and our dedication to finding a better way to conduct business effectively and efficiently. Our mission is to develop best-of breed solutions that encompass scalability, flexibility, and ease-of-use without losing focus of being affordable.



In 1998, William Acme founded Acme Services — based on the philosophy that services should be simple and affordable. Today, Acme is based on the same principle.

Acme released its first product, a services tool that allows user easily. Because and use, sales c with offices in A United Kingdom tool has become partnerships have been made with every m

As a result, Mr. Acme and his team have ha with and listen to thousands of service prov were using Acme's tool, as well as what cha most valued in a services solution. Based o developed its first services managment platform solution built around Acme's

Edit  
Preview  
View Content Difference  
View History  
Add Task  
Properties  
Workarea  
Logout

## Content Block Examples

Ektron CMS400.NET has two types of content blocks.

Type	Displays	More information
Static	One content block on the Web page	"Static Content Block" on page 612
Dynamic	A content block whose ID is passed through the URL	"Dynamic Content Block" on page 612

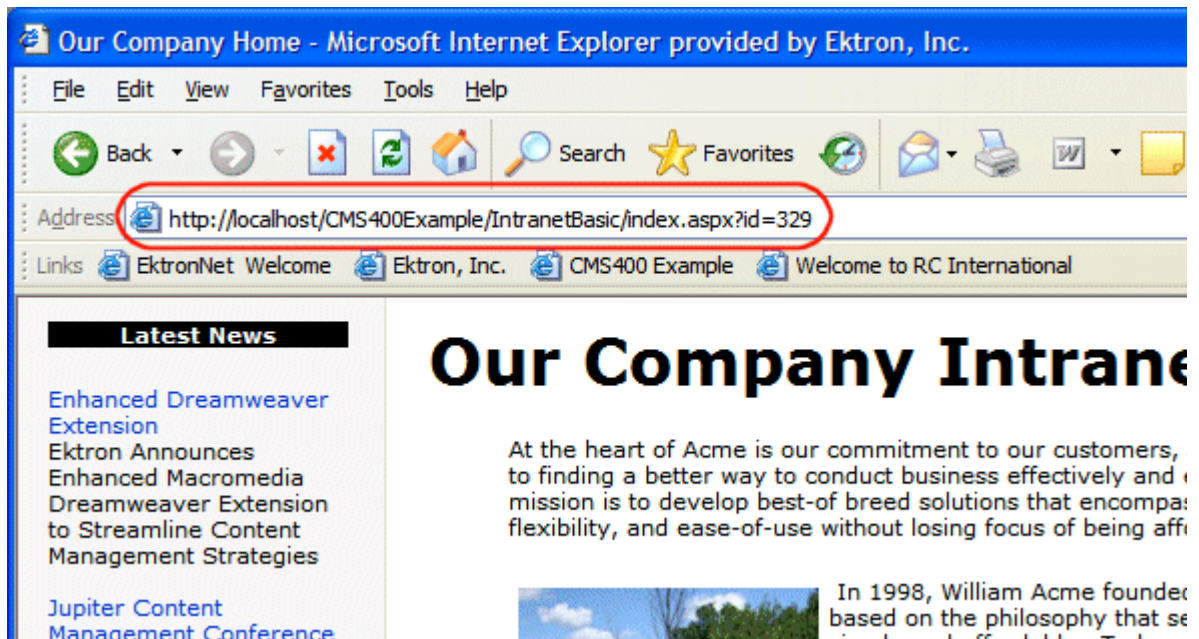
In addition, a content block can contain XHTML or XML content. For more information, see "XML Content Block" on page 614.

## Static Content Block

The following illustrates how to insert a static content block on a template.

```
<tr>
 <td>
 <% ecmContentBlock(329) %>
 </td>
</tr>
```

In the example above, the function retrieves the content block with an id of 329 from the database to the browser.

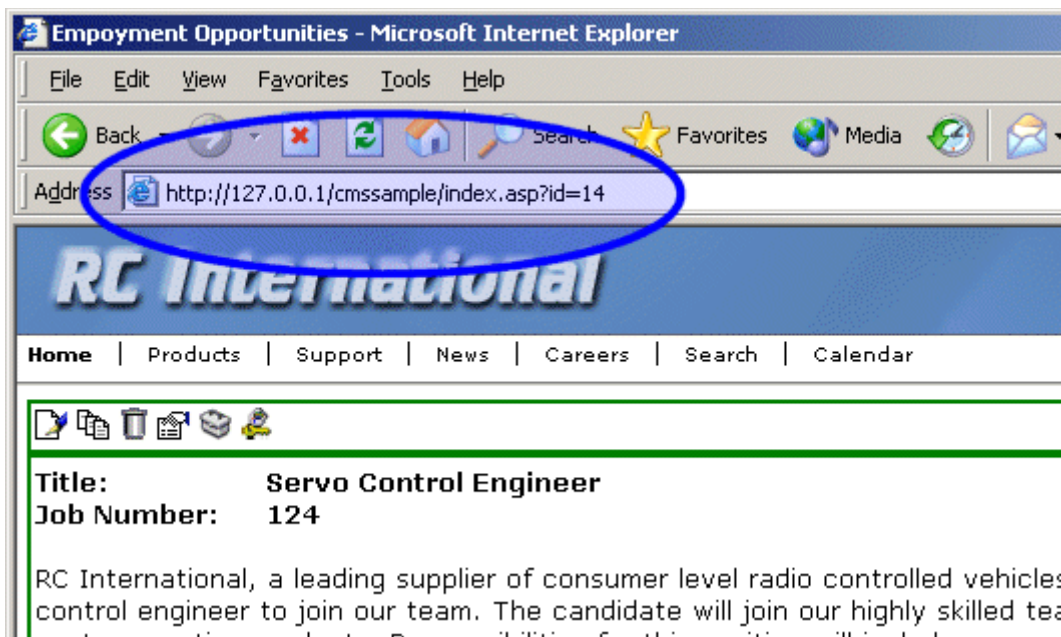


## Dynamic Content Block

The following example shows the content block tag where the id is being passed as a URL parameter. Use this format with dynamic template.

```
<tr>
 <td>
 <% ecmContentBlock(request.QueryString("id")) %>
 </td>
</tr>
```

When a user accesses this page, it adds the content block ID number to the tag's template, and the content block with the corresponding ID appears on the Web page in the specified template.



The `id=` convention is used by **Ektron CMS400.NET** to generate quicklinks when content blocks are created. (For more information, see the **Ektron CMS400.NET** Administrator manual section "Managing Content" > "Forming URLs For Ektron CMS400.NET Documents.")

The dynamic `ecmContentBlock` function is almost the same as the static one (see "[Static Content Block](#)" on page 612). The only difference is the parameter, which makes the function dynamic. The parameter uses a query string call (`Request.QueryString`) to read the URL from the browser's address bar. It treats everything after the question mark as a list of key/value pairs separated by ampersands. So, `Request.QueryString` passes the key as a parameter and returns its associated value. For example, in URL `<http://localhost/cms400Min/index.asp?id=1&LangType=1036>`, the following `QueryString` calls return these values.

QueryString Call	Return value
<code>Request.QueryString("id")</code>	1
<code>Request.QueryString("LangType")</code>	1036

## XML Content Block

Here is an example of displaying an XML content block. Note that you specify a content block and an XSLT, which determines how the content block is displayed.

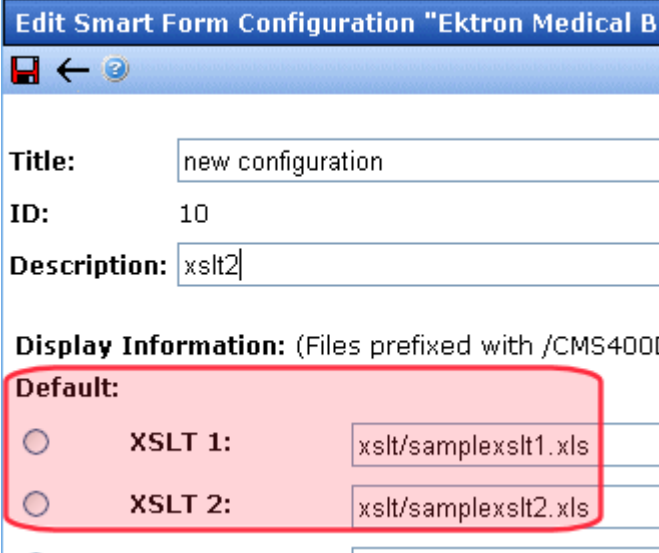
```
<tr>
 <td>
 <% ecmContentBlockEx 13, "", 1 %>
 </td>
</tr>
```

This example displays the content block ID=13, using XSLT1 as the display XSLT.

### ***The Function's Arguments***

The `ecmContentBlockEx` takes three arguments. The first argument identifies the content block to display. The second one is optional and, if used, specifies an external XSLT file.

If the second argument does not exist, the third argument specifies an XSLT identified in the Edit Smart Form screen. The following table provides additional detail about the second and third arguments.

To use this display XSLT	Enter this for argument 2	Enter this for argument 3
<p>The default XSLT specified in the Edit Smart Form screen (illustrated below). Note that, in this example, the <b>XSLT Packaged</b> option is the default XSLT since it is selected.</p> 	""	none
<p><b>XSLT Packaged:</b> from the Edit Smart Form screen (that is, the XSLT created in the Data Designer)</p> <hr/> <p>Note: If an XSLT package has not been created, and zero (0) is the third argument, the default XSLT specified in the Edit Smart Form screen is used.</p> <hr/>	""	0
<b>XSLT 1</b> from the Edit Smart Form screen	""	1
<b>XSLT 2</b> from the Edit Smart Form screen	""	2
<b>XSLT 3</b> from the Edit Smart Form screen	""	3

To use this display XSLT	Enter this for argument 2	Enter this for argument 3
An absolute or relative path to an XSLT	<p>An external XSLT file not specified in the Edit Smart Form screen.</p> <p>For example</p> <pre>&lt;% ecmContentBlockEx 13, "samplexslt. xsl"%&gt;</pre>	<p>does not matter - if a value exists in argument 2, argument 3 is ignored</p>

See Also: The "Managing XML" chapter of the **Ektron CMS400.NET** Administrator manual

## Dynamic and Static Combination

Here is an example of how you can use the two different content tags in the same table cell by using an IF statement

```
<tr>
 <td>
 <% if request.QueryString("id")<> "" then
 ecmContentBlock(request.QueryString("id"))else
 ecmContentBlock(14) end if %>
 </td>
</tr>
```

In this sample code, when a person accesses this page without passing through the content's ID number, the content with an ID=14 will be displayed. If you pass a content id through the URL like **index.asp?id=4**, the content block ID=4 will be displayed.

# Add Content

The add content function allows you to programmatically add a new content block to Ektron CMS400.NET without the need of logging in.

This function can be for several applications including, but not limited to:

- Applied to a Web form, allowing you to collect data from users about the content block
- Creating a loop function that moves information from a database to an Ektron CMS400.NET database

Here is the format for the add content function:

```
<% AddContent Username,
 Password,
 Domain,
 content_title,
 content_comment,
 ContentHtml,
 SummaryHTML,
 ContentLanguage,
 FolderID,
 GoLive,
 EndDate,
 MetaInfo %>
```

**IMPORTANT!** Every string that you pass to the function must be HTML encoded. Here is the VB.Net syntax for encoding a string.

```
myTitle = "John's Book"
myTitle = Server.HtmlEncode(myTitle)
```

Explained in the table below are the attributes for the add content block function.

Attribute	Description
Username	The username of the user to be authenticated for permissions to see whether the user is able to add a content block.
Password	The password for the username being authenticated.
Domain	If using Active Directory, specify the domain needed to authenticate the user.
Content_title	Specify a title for the content block being added.

Attribute	Description
Content_comment	Specify a historical comment for the content being added.
ContentHTML	Actual content for the content block.
SummaryHTML	Summary information about the content block.
ContentLanguage	Language of content.
FolderID	ID number of the content folder where the content block will be added to.
GoLive	Date and time the content will go live.  <a href="#">Must match date/time format set in the Web.config key ek_dateFormat.</a>
EndDate	Date and time the content will expire.  <a href="#">Must match date/time format set in the Web.config key ek_dateFormat.</a>
MetalInfo	Specify the information for the metadata which will be used as a reference point for finding other words or information. An example is: <pre>&lt;metadata&gt;&lt;meta id="3"&gt;Title&lt;/meta&gt;&lt;/metadata&gt;</pre> Where ID is the metadata definition ID whose value is being set. The value is set between the meta tags and the meta tag is repeatable for as many metadata definitions as you want to set for the given content block.

## Add Content Example

The following is an example of the add content function in a template:.

```
<tr>
 <td>
 <% AddContent "admin", "admin", "", "Content Title", "Comment", "This is the
 body of the content.", "This is the <i>summary</i> for the content.", 1033, 31,
 "06-14-2004 11:55 AM", "", "<metadata><meta id="3">ContentTitle</meta></
 metadata>"," %>
 </td>
</tr>
```



The previous example has the following properties:

- The user “admin” is used to be authenticated.
- The title of the new content block is “Content Title”.
- The comment for the content is “Comment”.
- The body of the content is “This is the **body** of the content”.
- The summary of the content block is “This is the *summary* for the content.
- The language ID = 1033, or English.
- The folder ID = 31 is where the content is added.
- The start date is 06-14-2004”.
- No end date is specified.
- The metadata definition ID being set is “3” and the value is ContentTitle.

When this function is executed, the content is added to Ektron CMS400.NET.  
Displayed here is the record of the content in the workarea.

**View Content "Content Title"**

View: English (U.S.) | Add: -select language-

**Properties** | Content | Summary | Meta Data | Alias | Comment | Tasks

**Properties**

<b>Content Title:</b>	Content Title
<b>Content ID:</b>	381
<b>Content Language</b>	English (U.S.)
<b>Status:</b>	Approved (Published)
<b>Last User To Edit:</b>	Application Administrator
<b>Last Edit Date:</b>	8/3/2005 12:27:47 PM
<b>Start Date:</b>	[None Specified]
<b>End Date:</b>	[None Specified]
<b>Action on End Date</b>	[None Specified]
<b>Date Created:</b>	8/3/2005 12:27:44 PM
<b>Approval Method:</b>	Do Not Force All Approvers
<b>Approvals:</b>	[None Specified]
<b>XML Configuration:</b>	[None Specified] HTML Content Assumed(inherited)
<b>Path</b>	\

Local intranet

# Search Functions

The search tag is used to display a text box, with a search button on the template. When seen on the template of a Web page, users can enter text and click on the search button to search through all the HTML content on the Web site. To create a search, Ektron has created two search functions:

- ecmSearch
- ecmSearchNoFormTag

Both functions have the same parameters and perform the same search. The only difference is ecmSearch automatically inserts form tags, ecmSearchNoFormTag does not.

The following is an example of form tags.

```
<form name="ecmsearch" method="post" action='search-
display.asp'>
```

```
</form>
```

When using ecmSearch, form tags are added automatically. When using ecmSearchNoFormTag you need to add the tags manually

When using the search function, ecmSearch and ecmSearchNoFormTag can be interchanged. Shown here is the format for the ecmSearch and ecmSearchNoFormTag:

<pre>&lt;% ecmSearch or ecmSearchNoFormTag</pre>	<pre>    "Starting Folder",     Recursive,     "Target Page",     Text Box Size,     Maximum Characters,     "Button Image Source",     "Button Text",     "Font Face",     "Font Color",     "Font Size",     Horizontal,     "Spare"</pre>	<pre>%&gt;</pre>
--------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------

The following table provides information about all the attributes for the search tags.

Attribute	Description	Required ?	For more information, see
Starting Folder	The folder you wish to search.	Yes	<a href="#">"Starting Folder" on page 624</a>
Recursive	Whether to search subfolders. 0=No, 1 = Yes	Yes	<a href="#">"Recursive" on page 624</a>
Target Page	The Web page to output the search results.	Yes	<a href="#">"Target Page" on page 624</a>
Text Box	The size of the text box for user input.	Yes	<a href="#">"Text Box" on page 625</a>
Maximum Characters	The maximum number of characters that the input text box can accept.	No	<a href="#">"Maximum Characters" on page 625</a>
Button Image Source	The image location for the search button. Blank ("") = standard submit button.	No	<a href="#">"Button Image Source" on page 626</a>
Button Text	The text used for the button if the standard submit button is chosen, or alternative text for the button if an image is chosen.	No	<a href="#">"Button Text" on page 627</a>
Font Face	The font of the search tag text.	No	<a href="#">"Font Face" on page 627</a>
Font Color	The font color of the search tag text. Blank ("") = Black.	No	
Font Size	The font size of the search tag text.	No	<a href="#">"Font Size" on page 629</a>
Horizontal	Determines if the search options are in a row or a column. 1 = horizontal, 0 = vertical	Yes	<a href="#">"Horizontal" on page 630</a>
Spare1	Currently not used.	Must be ""	<a href="#">"Spare 1" on page 630</a>

## Search Tag Example

The following example places a text field and a search button in a table cell inside an ASP template.

```
<tr>
 <td>
 <% ecmSearch "\",1,"search-
 display.asp",25,200,"","Search","Verdana",
 "#808080","2",0,"" %>
 </td>
</tr>
```

The previous example has the following properties:

- Searches the root folder, in this case the \content folder.
- Search is recursive.
- The target page is "search-display.asp".
- The size of the text box is 25.
- The search text box allows up to 200 characters.
- No image is used for a button. Instead, a standard submit button is used.
- The text on the submit button is "Search".
- The search options appear in Verdana with a color of #808080 and a size of 2pt.
- The search options are arranged vertically.

When added to a template, and visited, the search tag will appear similar to the following:

[News](#) | 
 [Careers](#) | 
 **[Search](#)** | 
 [Calendar](#)

## Search Tag Attributes

The search tag's attributes are described below.

### Starting Folder

In this attribute, specify the content folder where the search begins. There are two options for this attribute.

Value	Search begins in
"\"	root folder
"\folder path"	specified folder

### Recursive

When you setup the search to be recursive, you are allowing the search to search all the sub-folders associated with the starting folder you had specified. You have two options for this attribute:

Value	Search begins in
1	Recursive search (will search sub-folders of the starting folder).
0	Non-Recursive search (will only search starting folder).

### Target Page

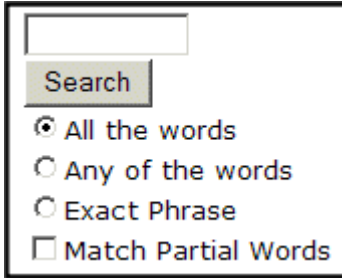
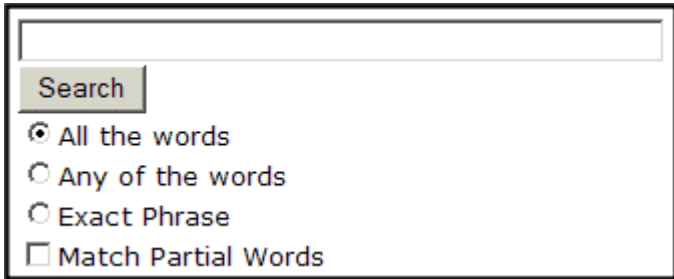
The page that displays the search results. You may enter any form of a URL for this page.

**NOTE** The target page *must* include the `searchdisplay` function to display search results.

To use the same page, enter a pound sign (#) into the field.

## Text Box

This attribute sets the size of the search text box. It can be any number, depending on the size of the search text box that you want.

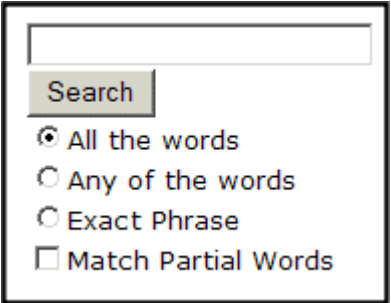
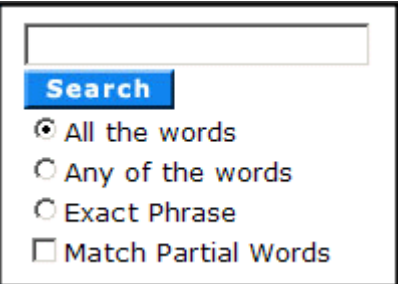
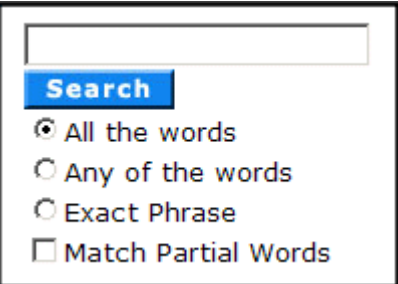
Text Box Size	Web Page View
10	
50	

## Maximum Characters

This attribute determines the maximum number of characters that a user can enter for search text.

## Button Image Source

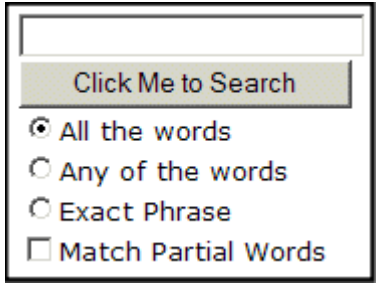
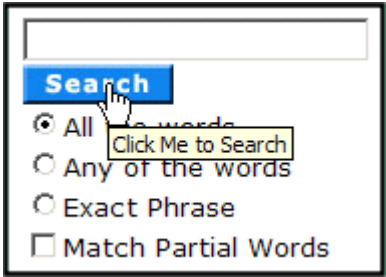
To use an image for the search button, enter the image's filename and path.

Button Image Source	Web Page View
""	
"search.jpg"	
"http://www.example.com/images/search.jpg"	



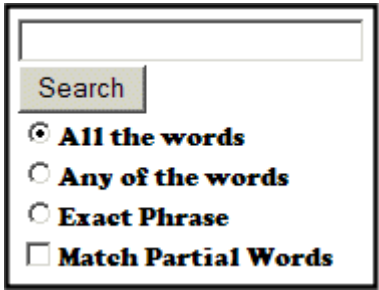
## Button Text

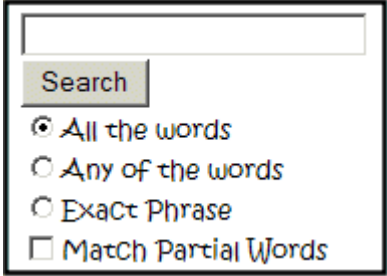
If no search button image is specified, the text in this field appears on the button.  
If an image is being used for the search button, this text is the image's alt text.

Button Text	Web Page View
"Click Me to Search"	
"Click Me to Search"	

## Font Face

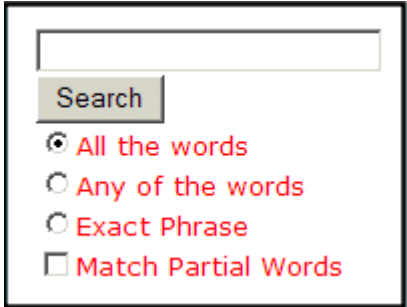
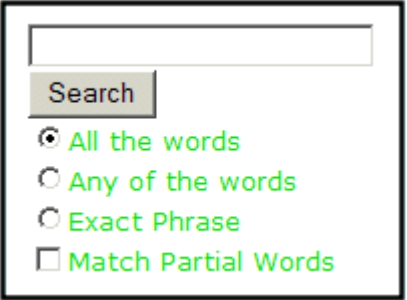
This font is used for the search options displayed on the Web page.

Font Face	Web Page View
Cooper Black	

Font Face	Web Page View
Kristen ITC	 <p>The image shows a web page view of a search interface. It includes a search input field, a 'Search' button, and four radio button options: 'All the words' (selected), 'Any of the words', 'Exact Phrase', and 'Match Partial Words'. The text is rendered in the 'Kristen ITC' font.</p>

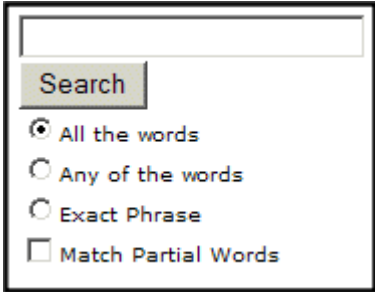
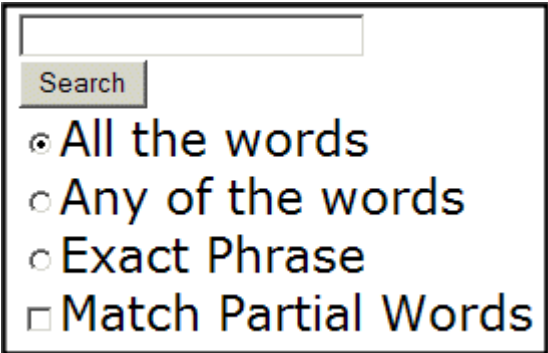
## Font Color

The color of the text in the search tag. You may enter the color name or a hexadecimal value of the color.

Font Color	Web Page View
Red	 <p>The image shows a web page view of a search interface where the text is colored red. It includes a search input field, a 'Search' button, and four radio button options: 'All the words' (selected), 'Any of the words', 'Exact Phrase', and 'Match Partial Words'.</p>
#14DA14	 <p>The image shows a web page view of a search interface where the text is colored green (#14DA14). It includes a search input field, a 'Search' button, and four radio button options: 'All the words' (selected), 'Any of the words', 'Exact Phrase', and 'Match Partial Words'.</p>

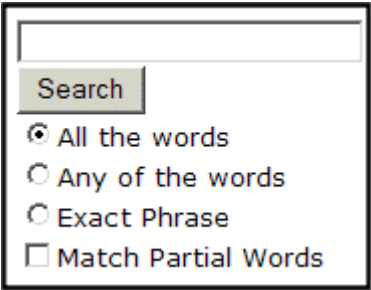

## Font Size

This is the size of the text on the search tag.

Font Size	Web Page View
1	 A screenshot of a search form with a font size of 1. The form includes a text input field, a 'Search' button, and four radio button options: 'All the words' (selected), 'Any of the words', 'Exact Phrase', and 'Match Partial Words' (checkbox). The text is very small.
5	 A screenshot of the same search form with a font size of 5. The text is significantly larger and more legible than in the font size 1 example.

## Horizontal

This attribute determines whether the search options are displayed vertically or horizontally.

Attribute	Web Page View
0 (vertical)	 A screenshot of a search interface where the search options are displayed vertically. It includes a search input field, a 'Search' button, and four radio button options: 'All the words' (selected), 'Any of the words', 'Exact Phrase', and 'Match Partial Words' (checkbox).
1 (horizontal)	 A screenshot of a search interface where the search options are displayed horizontally. It includes a search input field, a 'Search' button, and two radio button options: 'All the words' (selected) and 'Any of the words'.

## Spare 1

At the end of the Search tag is a attribute called Spare 1. This attribute has not yet been assigned an function. As of now, when you enter a search tag, you must pass this attribute "".

# Search Display Function

The search display custom tag creates the area where search results appear after a search is executed.

Here is the format of the ecmSearchDisplay tag.

```
<% ecmSearchDisplay "MaxNumberReturn",
 "StyleInfo",
 ShowDate %>
```

Below is a description of the tag's attributes.

Attribute	Description	Required	For more information, see
Max Number of content blocks Returned	Maximum number of content blocks returned. (0=Unlimited)	Yes	<a href="#">"Max Number of Content Blocks Returned" on page 632</a>
Style Info	An HTML style string used within the HTML "hyperlink" generated by the search output. Can control font, color, size, etc.	No	<a href="#">"Style Info" on page 633</a>
ShowDate	Displays the last modified date of the content block. 0=No 1=Yes	No	<a href="#">"Show Date" on page 634</a>

## Search Display Example

Insert the following on the page that displays search results. You may place it on the same page as the search tag or a separate page.

```
<tr>
 <td>
 <% ecmSearchDisplay 0,"","1" %>
 </td>
</tr>
```

This search display example has the following properties:

- Returns unlimited results when the search is performed.
- Contains no style information.
- Shows the last modified date of the content.

## Search Display with Default Content

You can also set up the search display tag to display a default content block if a search is not performed. Then, if a search is performed, the results replace the content block. The sample code below does that:

```
<tr>
 <% if (Request.Form("ecmsearchtype") <> "") then %>
 <td>
 <% ecmSearchDisplay 0,"" %>
 </td>
 <% else %>
 <td>
 <% ecmContentBlock(1) %>
 </td>
 <% end if %>
</tr>
```


## Search Display Attributes

Listed below are the attributes for the SearchDisplay tag.

### Max Number of Content Blocks Returned



This attribute determines the number of search results displayed on a page. You may enter any integer. To display unlimited results, enter 0 (zero).

Max Returns	Web Page View
0	<div> <b>Search Results</b>  <a href="#">RC Sportster</a>  <a href="#">RC Redstar</a>  <a href="#">RC Lilly</a>  <a href="#">RC Cheetah</a>  <a href="#">Products Page</a>  <a href="#">RC International Ships RC Sportster</a>  <a href="#">RC International Hosts RC Conference</a>  <a href="#">RC International Announces the RC Cheetah</a> </div>

Max Returns	Web Page View
3	

## Style Info

An HTML style string used within the HTML “hyperlink” generated by the search output. Can control font, color, size, etc.

Style Info	Web Page View
""	
"font-family:arial; font-weight:bold; background-color:#cccccc; border:solid blue 1pt; margin:2px; width:50%;"	

## Show Date

This attribute determines if the content block's last modified date appears next to the title. This is useful when determining which content block is the most recent. You have two choices.

Attribute	Web Page View
0	<div data-bbox="358 380 965 729"> <p><b>Search Results</b></p> <p>RC Sportster  RC Redstar  RC Lilly  RC Cheetah  Products Page  RC International Ships RC Sportster  RC International Hosts RC Conference  RC International Announces the RC Cheetah</p> </div>
1	<div data-bbox="358 789 1310 1130"> <p><b>Search Results</b></p> <p>RC Sportster (8/27/2002 7:52:40 PM)  RC Redstar (8/27/2002 7:52:01 PM)  RC Lilly (8/27/2002 7:51:26 PM)  RC Cheetah (8/27/2002 7:50:53 PM)  Products Page (8/27/2002 7:50:09 PM)  RC International Ships RC Sportster (8/27/2002 2:46:12 PM)  RC International Hosts RC Conference (8/27/2002 2:44:44 PM)  RC International Announces the RC Cheetah (8/27/2002 2:43:01 PM)</p> </div>



# List Summary Function

The list summary function is used to display the list of content summaries on a Web page.

Shown here is the format of the list summary function:

```
<% ecmListSummary "Folder",
 Recursive,
 ShowSummary,
 "StyleInfo",
 "OrderBy, Asc/Desc",
 "ShowInfo",
 MaxNumber,
 "ObjType",
 "OptionList" %>
```

A list and description of the attributes are shown in this table:

Attribute	Description	Options
Folder	This is the folder the summary information will come from.	Any folder
Recursive	Allow the tag to get summary info from the child folders.	0 = Not Recursive 1 = Recursive
ShowSummary	Do you want the content summary to be displayed?	0 = No 1 = Yes
StyleInfo	Inline style information for the generated text hyperlink.	Any string
OrderBy, ASC/DESC	Order the hyperlinks by:	"Title" "DateModified" "DateCreated" "Startdate" "AuthorFname" "AuthorLname"
	Order the hyperlinks in ascending or descending order.	"Title,asc" "Title,desc"

Attribute	Description	Options
ShowInfo	Information that will appear to the right of the hyperlink.	"DateModified" "DateCreated" "AuthorFname" "AuthorLname"
MaxNumber	Maximum number of summaries returned.	0 = Unlimited
ObjType	<p>Determines whether the list summary is for the folder or a single piece of content.</p> <p>For example:</p> <p>If the folder property = \books and "Folder" is chosen as the ObjType, the list summary for the folder with books appears.</p> <p>If the folder property = 1 and you choose "Content" as the ObjType, a single summary for content with an id = 1 appears.</p> <p>See Also: <a href="#">"Single Summary Function" on page 648</a></p> <p>The default ObjType is "Folder" when left blank.</p>	"" "Folder" "Content"
OptionList	Not yet implemented	MUST BE ""

## List Summary Example

Shown below is an example of the ListSummary function in an ASP Web page.

```
<TR>
 <TD>
 <% ecmListSummary "\Marketing\News",0,1,"",
 "Title,asc","DateModified",0,"","" %>
 </TD>
</TR>
```

The previous example has the following properties:

- Displays summaries from the \Marketing\News folder.
- Not recursive.
- The summaries will be shown.
- No style was applied to the hyperlinks.
- The results will be displayed in ascending order by title.
- The date modified of each content block will be displayed.

- Unlimited results will be shown.

Here is how the summaries will appear on the web page.



## List Summary Attributes

Listed below are all the different attributes and descriptions about them.

### Starting Folder

In this attribute, you will need to specify which content folder you will want to get all the summaries from. There are two options for this attribute:

Value	Description
""	By leaving the value blank, all the summaries from the root content folder will be displayed.
"\Folder Path"	You can specify a specific folder to display the summaries from by inserting the folder path

## Recursive

When you setup the search to be recursive, you are allowing the search to search all the subfolders associated with the starting folder you had specified. You have two options for this attribute:

Value	Description
1	Recursive - summaries in the starting folder's subfolders will also be displayed.
0	Non-recursive - only summaries in the starting folder will be displayed.

## Show Summary

The show summary attribute displays or suppresses the summary in the summary list. You have two options for this attribute.

Attribute	Web Page View
0	<a href="#">RC International Ships RC Sportster</a> <a href="#">RC International Hosts RC Conference</a> <a href="#">RC International Announces the RC Cheet</a>
1	<a href="#">RC International Announces the RC Cheetah (8/27/2002 2:43:01 PM)</a> RC International, a leading RC racing company, has announced the availability of its new flagship product, the RC Cheetah.  <a href="#">RC International Hosts RC Conference (8/27/2002 2:44:44 PM)</a> RC International, the fastest growing company in the RC industry, announces plans to host the 3rd annual RC conference in Atlanta, Georgia.

## Style Info

In this field, you may enter inline style information for the generated text hyperlinks displayed on the screen.

Style Info	Web Page View
<pre> </pre>	<p><a href="#">RC International Announces the RC Cheetah (8/27/2002 2:43:01</a> RC International, a leading RC racing company, has announced the availability of its new flagship product, the RC Cheetah.</p> <p><a href="#">RC International Hosts RC Conference (8/27/2002 2:44:44 PM)</a> RC International, the fastest growing company in the RC industry, announces plans to host the 3rd annual RC conference Atlanta, Georgia.</p>
<pre> "font-family:arial; font-weight:bold; background-color:#cccccc; border:solid blue 1pt; margin:2px; width:100%;" </pre>	<p><b><a href="#">RC International Announces the RC Cheetah</a></b> RC International, a leading RC racing company, has announced the availability of its new flagship product, the RC Cheetah.</p> <p><b><a href="#">RC International Hosts RC Conference</a></b> RC International, the fastest growing company in the RC industry, announces plans to host the 3rd annual RC conference Atlanta, Georgia.</p>

## Order By

This attribute will sort the summary results by different options. There are five different ways to sort the summaries:

Value	Summaries sorted by
Title	title
DateModified	date modified
Date Created	date created
Startdate	The Start Date assigned to the content block when created.
AuthorFname	author's first names

Value	Summaries sorted by
AuthorLname	author's last name

### Ascending/Descending

This attribute is an addition to the previous OrderBy attribute. By default, the list of the summaries are descending, meaning they will follow the path of A-B-C, or 1-2-3. When set to ascending, they will order by Z-Y-X or 3-2-1. You will have the option to re arrange these to display either in ascending or descending order.

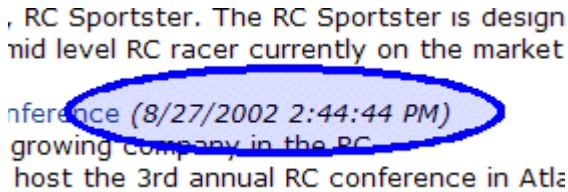
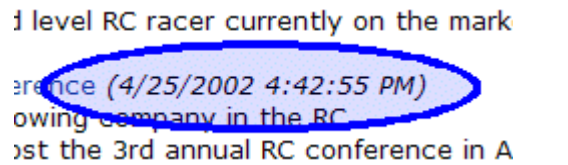
To do this, add the following to the OrderBy attribute:

Value	Description
asc, 1	This will order the Summaries in an ascending order
desc, 0	This will order the Summaries in a descending order. This is the default setting.

By using the ascending/descending option, you will have the option to set the order of content blocks from most recent to oldest, or vice versa. As well as ordering the list by the user's first names or last names in ascending or descending order.

### Show Info

The show info field determines what sort of information is displayed to the right of the hyperlinked title. There are four different options that you may choose from:

Value	Description	Web Page View
DateModified	Displays the date the content was last modified.	
***DateCreated	Displays the date when the content was first created.	

Value	Description	Web Page View
AuthorFname	Displays the first name of the user who last edited the content block.	mid level RC racer currently ference (John) growing company in the R host the 3rd annual RC cc
AuthorLname	Displays the last name of the user who last edited the content block.	ct, RC Sportster. The RC Sp d mid level RC racer currentl Conference (Doe) st growing company in the R to host the 3rd annual RC cc

## Max Number

This attribute determines how many summaries appear on a page. Enter the number of summaries to be displayed. To display all summaries, enter 0 (zero) and you will display unlimited results.

# XML List Summary Function

The XML List Summary function wraps XML around the properties of a list of content that's created as a list summary in the Ektron CMS400.NET Workarea. Once created, you can use the XML to display the list summary information. The ecmListSummaryXML function allows you to easily define how the list summary data appears on the Web page.

Shown here is the format of the ecmListSummaryXML function.

```
<% ecmListSummaryXML "Folder",
 "Recursive",
 "OrderBy",
 "MaxNumber",
 "ObjType",
 "SummaryType" %>
```

The ecmListSummaryXml function attributes are explained below.

Attribute	Description	Options
Folder	This is the folder the summary information will come from. See Also: <a href="#">"Starting Folder" on page 637</a>	Any folder
Recursive	Allow the tag to get summary info from the child folders. See Also: <a href="#">"Recursive" on page 638</a>	0 = Not Recursive 1 = Recursive
OrderBy, ASC/ DESC	Order the hyperlinks by: See Also: <a href="#">"Order By" on page 639</a>	"Title" "DateModified" "DateCreated" "Startdate" "AuthorFname" "AuthorLname"
	Order the hyperlinks in ascending or descending order.	"Title,asc" "Title,desc"
MaxNumber	Maximum number of summaries returned. See Also: <a href="#">"Max Number" on page 641</a>	0 (Zero) = Unlimited



Attribute	Description	Options
ObjType	<p>Determines whether the list summary is for the folder or a single piece of content.</p> <p>For example:</p> <p>If the folder property = \books and "Folder" is chosen as the ObjType, the list summary for the folder with books appears.</p> <p>If the folder property = 1 and you choose "Content" as the ObjType, a single summary for content with an id = 1 appears.</p> <p>See Also: <a href="#">"Single Summary Function" on page 648</a></p> <p>The default ObjType is "Folder" when left blank.</p>	<p>""</p> <p>"Folder"</p> <p>"Content"</p>
SummaryType	<p>Used as a filter, this property determines the type of content that appears in the list summary. The default is "Content" when left blank.</p>	<p><b>"AllTypes"</b> - displays all of the content types for the given folder.</p> <p><b>"Content"</b> - displays a list of content blocks.</p> <p><b>"Forms"</b> - forms appear in the list summary.</p> <p><b>"Archive_Content"</b> - archived content blocks appear in the list summary.</p> <p><b>"Archive_Forms"</b> - archived forms appear in the list summary.</p> <p><b>"Assets"</b> - assets, such as offices documents, appear in the list summary.</p> <p><b>"Archive_Assets"</b> - archived assets appear in the list summary.</p> <p><b>"LibraryItem"</b> - library items appear in the list summary.</p> <p><b>"NonLibraryContent"</b> - all types of content appear in the list summary except for library items.</p>

When using the XML list summary function, the following tags are created around the properties of each item in the list summary.

XML Tags	Description
<code>&lt;Content&gt;&lt;/Content&gt;</code>	Declares a new section of content in the XML collection.
<code>&lt;ID&gt;&lt;/ID&gt;</code>	The content block ID. For example: <code>&lt;ID&gt;27&lt;/ID&gt;</code>
<code>&lt;Title&gt;&lt;/Title&gt;</code>	The title of the content block. For example: <code>&lt;Title&gt;HTML for the World Wide Web with XHTML and CSS&lt;/Title&gt;</code>
<code>&lt;QuickLink&gt;&lt;/QuickLink&gt;</code>	The quicklink associated with the content block. For example: <code>&lt;QuickLink&gt;/cms400Min/collection.aspx?id=27&lt;/QuickLink&gt;</code>
<code>&lt;Teaser&gt;&lt;/Teaser&gt;</code>	The summary of the content block. For example: <code>&lt;Teaser&gt;The easiest HTML book weve seen that still manages to be comprehensive.&lt;/Teaser&gt;</code>
<code>&lt;StartDate&gt;&lt;/StartDate&gt;</code>	The content block's start date formatted as a .NET date type For example: <code>&lt;StartDate&gt;1/1/0001 12:00:00 AM&lt;/StartDate&gt;</code>
<code>&lt;DateModified&gt;&lt;/DateModified&gt;</code>	The last date the content block was modified. For example: <code>&lt;DateModified&gt;1/19/2005 4:18:06 PM&lt;/DateModified&gt;</code>
<code>&lt;EndDate&gt;&lt;/EndDate&gt;</code>	The date the content block stops running on the Web site For example: <code>&lt;EndDate&gt;12/31/9999 12:00:00 AM&lt;/EndDate&gt;</code>
<code>&lt;LastEditorFname&gt;&lt;/LastEditorFname&gt;</code>	The first name of the last person who edited the content block. For example: <code>&lt;LastEditorFname&gt;John&lt;/LastEditorFname&gt;</code>

XML Tags	Description
<code>&lt;LastEditorLname&gt;&lt;/LastEditorLname&gt;</code>	The last name of the last person who edited the content block. For example: <code>&lt;LastEditorLname&gt;Johnson&lt;/LastEditorLname&gt;</code>
<code>&lt;DisplayStartDate&gt;&lt;/DisplayStartDate&gt;</code>	The content block's start date. It is formatted as a string that represents Ektron CMS400's display of the date.
<code>&lt;FolderID&gt;&lt;/FolderID&gt;</code>	The ID of the folder in which the content block is contained. For example: <code>&lt;FolderID&gt;0&lt;/FolderID&gt;</code>
<code>&lt;ContentStatus&gt;&lt;/ContentStatus&gt;</code>	The status of the content block.
<code>&lt;Language&gt;&lt;/Language&gt;</code>	The language of the content block. For example: <code>&lt;Language&gt;1033&lt;/Language&gt;</code>
<code>&lt;DisplayDateModified&gt;&lt;/DisplayDateModified&gt;</code>	edited. It is formatted as a string that represents Ektron CMS400's display of the date. For example: <code>&lt;DisplayDateModified&gt;1/19/2005 4:18:06 PM&lt;/DisplayDateModified&gt;</code>
<code>&lt;DisplayEndDate&gt;&lt;/DisplayEndDate&gt;</code>	The content block's end date. It is formatted as a string that represents Ektron CMS400's display of the date. For example: <code>&lt;DisplayEndDate&gt;12/31/9999 12:00:00 AM&lt;/DisplayEndDate&gt;</code>

Here is how the raw XML list summary output appears before an XSLT is applied.

```
<Content>
 <ID>24</ID>
 <Title>ASP.NET Unleashed</Title>
 <QuickLink>/cms400Min/collection.aspx?id=24</QuickLink>
 <Teaser><p>ASP.NET Unleashed, Second Edition is really big, really thorough, and
really <i>good</i>.</p></Teaser>
 <StartDate>1/1/0001 12:00:00 AM</StartDate>
 <DateModified>1/12/2005 7:53:31 PM</DateModified>
 <EndDate>12/31/9999 12:00:00 AM</EndDate>
 <LastEditorFname>Application</LastEditorFname>
 <LastEditorLname>Administrator</LastEditorLname>
 <DisplayStartDate></DisplayStartDate>
 <FolderID>0</FolderID>
 <ContentStatus></ContentStatus>
 <Language>0</Language>
 <DisplayDateModified>1/12/2005 7:53:31 PM</DisplayDateModified>
 <DisplayEndDate></DisplayEndDate>
</Content>
<Content>
 <ID>25</ID>
 <Title>Microsoft ASP.NET Programming with Microsoft Visual C#.NET</Title>
 <QuickLink>/cms400Min/collection.aspx?id=25</QuickLink>
 <Teaser><p>Teach yourself how to write high-performance Web applications with ASP.NET and
Visual C# .NET--one step at a time.</p></Teaser>
 <StartDate>1/1/0001 12:00:00 AM</StartDate>
 <DateModified>1/4/2005 7:16:21 PM</DateModified>
 <EndDate>12/31/9999 12:00:00 AM</EndDate>
 <LastEditorFname>Application</LastEditorFname>
 <LastEditorLname>Administrator</LastEditorLname>
 <DisplayStartDate></DisplayStartDate>
 <FolderID>0</FolderID>
 <ContentStatus></ContentStatus>
 <Language>0</Language>
 <DisplayDateModified>1/4/2005 7:16:21 PM</DisplayDateModified>
 <DisplayEndDate></DisplayEndDate>
</Content>
```

# Archive List Summary Function

The Archive list summary function displays on a Web page a list of content summaries for archived content blocks. For example, you might have a current news page and an archived news page. Use the List Summary Function to display current news, and this function to display older news stories.

On the Content Block **schedule** screen, the user must set an **End Date** then choose the Archive and Display option to have the content blocks appear via this function.

Shown here is the format of the Archive List Summary function:

```
<% ecmListArchiveSummary "StartingFolder",
 Recursive,
 ShowSummary,
 "StyleInfo",
 "OrderBy, Asc/Desc",
 "ShowInfo",
 MaxNumber,
 "Spare1",
 "Spare2"
 %>
```

This function has the same parameters as the List Summary Function. See ["List Summary Function" on page 635](#) for details.

# Single Summary Function

The single summary tag is used to display the summary of one content block in the Ektron CMS400.NET Web site, instead of a list of all the summaries in a content folder. Seen below is the single summary tag:

```
<% ecmSingleSummary "ContentID",
 ShowSummary,
 "StyleInfo",
 "ShowInfo",
 "Spare1" %>
```

You can look at the single summary tag as just a simpler form of the List summary tag. The format for each of them is almost identical, but the single summary tag does not require as many attributes to be passed.

A list and description of the attributes are shown in this table:

Attribute	Description	Required?
Content ID	This is the ID number of the content block the summary information will come from.	Yes
ShowSummary	Do you want the content summary to be displayed?	Yes
StyleInfo	Inline style information for the generated text hyperlink.	No
ShowInfo	Information that will appear to the right of the hyperlink.	No
Spare1	Not yet implemented	MUST BE ""

## Single Summary Example

Shown below is an example of the single summary function in an ASP Web page.

```
<TR>
 <TD>
 <% ecmSingleSummary "14",1,"","DateModified","" %>
 </TD>
</TR>
```

The previous example had the following properties:

- Displays the summary for the content block with an ID=14

- The summary will be shown
- No style was applied to the hyperlinks
- The last date the content was modified will be displayed

Here is how the summary will appear on the web page.

Servo Control Engineer #124 (10/29/2002 6:51:02 PM)  
RC International is looked for a highly skilled servo control engineer to join our team.

## Single Summary Attributes

Listed below are all the different attributes and descriptions about them.

### Content ID

In this attribute, you will need to specify which content block you will want the summary to be displayed from.

### Show Summary

The show summary attribute will allow or disallow the summary to be displayed in the summary list. You have two options for this attribute:

Value	Web Page View
0	Servo Control Engineer #124
1	Servo Control Engineer #124 RC International is looked for a highly skilled servo control engineer to join our team.

# Style Info

In this field, you may enter inline style information for the generated text hyperlinks displayed on the screen.

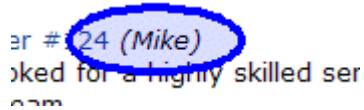
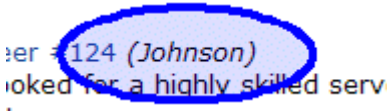
Style Info	Web Page View
""	<a href="#">Servo Control Engineer #124</a> RC International is looked for a highly skilled servo control engineer to join our team.
font-family:arial; font-weight:bold; background-color:#cccccc; border:solid blue 1pt; margin:2px; width:100%;	<div>Servo Control Engineer #124</div> <div>RC International is looked for a highly skilled servo control engineer to join our team.</div>

# Show Info

The show info field determines what sort of information is displayed to the right of the hyperlinked title. There are five different options that you may choose from:

Value	Displays the...	Web Page View
ID	ID number for the content block.	<a href="#">Servo Control Engineer #124 (14)</a> RC International is looked for a highly skilled servo control engineer to join our team.
DateModified	date the content was last modified.	<a href="#">Servo Control Engineer #124 (8/27/2002 2:40:26 PM)</a> RC International is looked for a highly skilled servo control engineer to join our team.
DateCreated	date when the content was first created.	<a href="#">Servo Control Engineer #124 (4/25/2002 4:54:42 PM)</a> RC International is looked for a highly skilled servo control engineer to join our team.



Value	Displays the...	Web Page View
AuthorFname	first name of the user who last edited the content block.	
AuthorLname	last name of the user who last edited the content block	

**Spare 1**

There is one attribute fields at the end that are currently not used. This is here for future scalability. When you insert a single summary tag into your template, you must leave this attributes as ""

# RSS Summary Function

The RSS summary function is used to create an XML list of content summaries formatted for RSS Syndication.

Shown here is the format of the RSS summary function:

```
<% ecmRssSummary "Folder",
 Recursive,
 "OrderBy, Asc/Desc",
 MaxNumber,
 %>
```

A list and description of the attributes are shown in this table:

Attribute	Description	Options
Folder	This is the folder the summary information will come from.	Any folder
Recursive	Allow the tag to get summary info from the child folders.	0 = Not Recursive 1 = Recursive
OrderBy, ASC/DESC	Order the hyperlinks by:	"Title" "DateModified" "DateCreated" "Startdate" "AuthorFname" "AuthorLname"
	Order the hyperlinks in ascending or descending order.	"Title,asc" "Title,desc"
MaxNumber	Maximum number of summaries returned.	0 = Unlimited

## RSS Summary Example

Shown below is an example of the ecmRssSummary function in an ASP Web page.

```
<TR>
 <TD>
 <% ecmRssSummary "\CMS400Min\", 0, "Title,asc", 0 %>
 </TD>
</TR>
```

The previous example has the following properties:

- Displays summaries from the \CMS400Min\ folder.
- Not recursive.
- The results will be displayed in ascending order by title.
- Unlimited results will be shown.

Here is how an RSS summary appears when viewed without an RSS reader on a web page.

```
<?xml version="1.0" ?>
- <rss xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" version="2.0">
- <channel>
 <title>example listsummary</title>
 <link>http://localhost/CMS400Min/developer/ListSummary/RSS_ListSummary.aspx?show=true</link>
 <description />
- <item>
 <title>eWebEditPro+XML V4.2 supports the vision of create content once, reuse many times</title>
 <link>http://192.168.0.82/CMS400Min/Developer/ListSummary/pr.aspx?id=290</link>
- <description>
 - <![CDATA[<p>Ektrons XML Authoring Tool Makes it Easier to Create Structured, Validated Content to
Support an Organizations Information and Content Management Strategies</p>]]>
 </description>
 <author>Application Administrator</author>
 <pubDate>3/7/2006 2:00:46 PM</pubDate>
 </item>
- <item>
 <title>Ektron Offers a Visual Development Environment for Rapid CMS Integration and Deployment</title>
 <link>http://192.168.0.82/CMS400Min/Developer/ListSummary/pr.aspx?id=282</link>
- <description>
 - <![CDATA[<p>Ektron leverages strong capabilities of Visual Studio.NET to give developers a truly visual
environment for integrating CMS components into Web templates</p>]]>
 </description>
 <author>Application Administrator</author>
 <pubDate>3/7/2006 2:02:19 PM</pubDate>
 </item>
- <item>
 <title>Ektron Announces Winner of All-Stars Customer Competition</title>
 <link>http://192.168.0.82/CMS400Min/Developer/ListSummary/pr.aspx?id=292</link>
- <description>
 - <![CDATA[<p>-- All-Stars competition identifies the most innovative and effective
implementations of Ektrons Web solutions and tools --</p>]]>
 </description>
 <author>Application Administrator</author>
 <pubDate>3/6/2006 10:51:52 AM</pubDate>
 </item>
</channel>
</rss>
```

## RSS Summary Attributes

Listed below are all the different attributes and descriptions about them.

### Starting Folder

In this attribute, you will need to specify which content folder you will want to get all the summaries from. There are two options for this attribute:

Value	Description
""	By leaving the value blank, all the summaries from the root content folder will be displayed.
"\Folder Path"	You can specify a specific folder to display the summaries from by inserting the folder path

### Recursive

When you setup the search to be recursive, you are allowing the search to search all the subfolders associated with the starting folder you had specified. You have two options for this attribute:

Value	Description
1	Recursive - summaries in the starting folder's subfolders will also be displayed.
0	Non-recursive - only summaries in the starting folder will be displayed.

### Order By

This attribute will sort the summary results by different options. There are five different ways to sort the summaries:

Value	Summaries sorted by
Title	title
DateModified	date modified
Date Created	date created
Startdate	The Start Date assigned to the content block when created.
AuthorFname	author's first names

Value	Summaries sorted by
AuthorLname	author's last name

### ***Ascending/Descending***

This attribute is an addition to the previous OrderBy attribute. By default, the list of the summaries are descending, meaning they will follow the path of A-B-C, or 1-2-3. When set to ascending, they will order by Z-Y-X or 3-2-1. You will have the option to rearrange these to display either in ascending or descending order.

To do this, add the following to the OrderBy attribute:

Value	Description
asc, 1	This will order the Summaries in an ascending order
desc, 0	This will order the Summaries in a descending order. This is the default setting.

By using the ascending/descending option, you will have the option to set the order of content blocks from most recent to oldest, or vice versa. As well as ordering the list by the user's first names or last names in ascending or descending order.

### **Max Number**

This attribute determines how many summaries appear on a page. Enter the number of summaries to be displayed. To display all summaries, enter 0 (zero) and you will display unlimited results.

# Collection Function

The collection function is used to display the list of links that were created as a collection in the Ektron CMS400.NET Workarea. The `ecmCollection` function is highly customizable, allowing you to easily define how the collection data appears on the Web page.

Shown here is the format of the `ecmCollection` function.

```
<% ecmCollection "id",
 "Display Function" %>
```

The `ecmCollection` function attributes are explained below.

Attribute	Description
id	The id of the collection that you would like to be displayed is defined here.
displayFunction	Specify a function that is defined in the API and determines how the appearance of the collection data on a Web page. The display function must also be defined in the <code>ecmCollection</code> function.

See Also: The **Ektron CMS400.NET** User Manual Section "Working with Collections" > "Comparison of Collections, Menus, and the List Summary Features"

## Collection Examples

The following are examples of the collection function in the Ektron CMS400.NET sample Web site. One example uses the `ecmNavigation` display function, and the other uses the `ecmTeaser` display function.

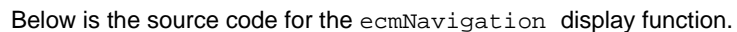
The `ecmNavigation` and `ecmTeaser` display functions are sample display functions included with Ektron CMS400.NET.

### The `ecmNavigation` Display Function

The following sample collection function is being used as a navigation menu in the Ektron CMS400.NET sample Web site.

```
<% ecmCollection 1, "ecmNavigation" %>
```

In this example, the collection with an ID=1 is displayed, and the `ecmNavigation` function defines the appearance of the collection data on a Web page.



As you can see, the `ecmNavigation` function is a simple ASP function that creates a table of the links to the specified collection. It displays the title of the content blocks by using the `info("ContentTitle")` building block.

The following is an example of a collection function being used as a navigation menu in the Ektron CMS400.NET sample Web site.

*Ektron CMS400.NET Developer Reference Manual, Version 7.0, Rev 8*



In this example, the collection with an ID=2 is displayed, and the `ecmTeaser` function defines the appearance of the collection data on the Web page. `ecmTeaser` displays the title and summary of each content block in the collection. Here is how the collection might appear on a Web page.



The format of the collection on the Web page depends on the `displayFunction` that is used. Below is the source code for the display function `ecmTeaser`.

```
Function ecmTeaser(cInfo)
 dim html, info
 html = "<table border=""0"" cellspacing=""0"" cellpadding=""0""
 width=""100%"">"
 for each info in cInfo
 html = html & "<tr><td><a href=""
 html = html & CStr(info("ContentLinks"))
 html = html & "" "" >"
 html = html & Cstr(info("ContentTitle"))
 html = html & " ("
 html = html & info("DateModified")
 html = html & ")</td></tr><tr><td>"
 html = html & Cstr(info("ContentTeaser"))
 html = html & "</td></tr><tr><td> </td></tr>"
 next
 html = html & "</table>"
 ecmTeaser = html
end Function
```

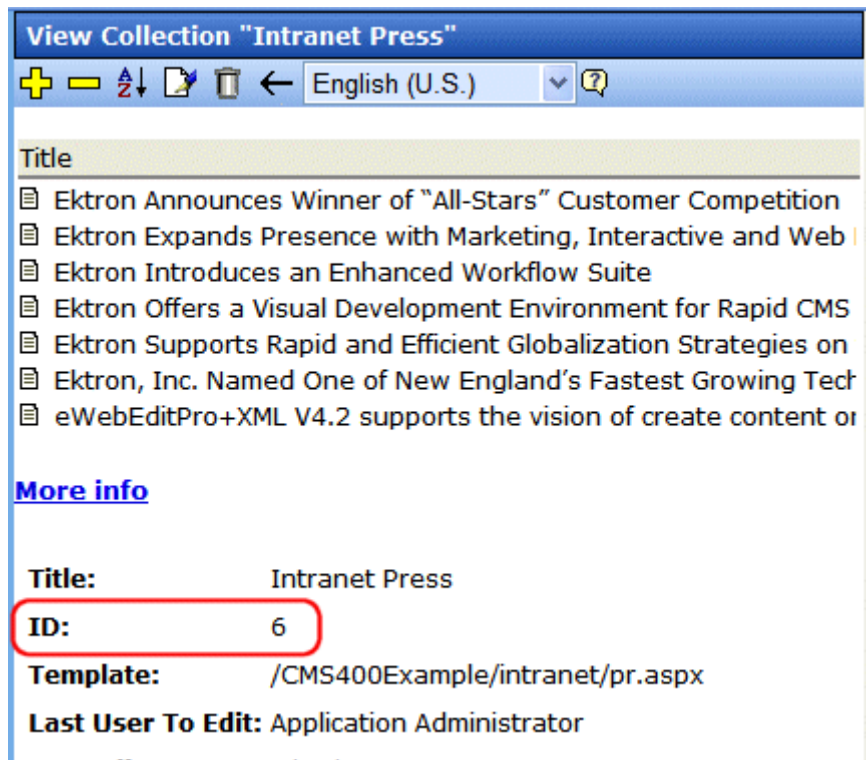
As you can see, the `ecmTeaser` function is an ASP function that creates a table of links to the specified collection. The `ecmTeaser` function displays the:

- Content title
- Date modified
- Content block summary

## Collection Attributes

### ID

The collection ID number is assigned to the collection when it is created.



### DisplayFunction

A display function determines which elements of the content blocks in a collection appear on a Web page. Via ASP scripting, you can create display functions or use any of the three included functions. You must define the display function in two places.

- In the ecmCollection function as a case
- In the API file as a function

# RSS Collection Function

The RSS collection function is used to create an XML list of links that were created as a collection in the Ektron CMS400.NET Workarea. The ecmRssCollection function is highly customizable, allowing you to easily define how the collection data appears on the Web page.

Shown here is the format of the ecmRssCollection function.

```
<% ecmRssCollection "id" %>
```

The ecmRssCollection function attributes are explained below.

Attribute	Description
id	The id of the collection that you would like to be displayed is defined here.

See Also: The **Ektron CMS400.NET** User Manual Section "Working with Collections" > "Comparison of Collections, Menus, and the List Summary Features"

Here is how an RSS Collection appears when viewed without an RSS reader on a web page.

```

<?xml version="1.0" ?>
<rss xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" version="2.0">
 <channel>
 <title>Books</title>
 <link>http://localhost/cms400Min/WorkArea/WebServices/ContentWS.asmx</link>
 <description></description>
 <item>
 <title>Visual Basic.NET How to Program</title>
 <link>http://localhost/cms400Min/collection.aspx?id=23</link>
 <description>Visual Basic .NET How to Program, Second Edition is an awesome resource for
the beginning VB.NET developer who is serious about the craft of programming. In 1,500 pages, this book
covers an incredible amount of territory, from the evolution of programming languages and web
development all the way to advanced data structures and web services.</description>
 <author>Application Administrator</author>
 <pubDate>3/3/2005 5:15:30 PM</pubDate>
 </item>
 <item>
 <title>Simply Visual Basic.NET 2003</title>
 <link>http://localhost/cms400Min/collection.aspx?id=26</link>
 <description>An Application-Driven Tutorial Approach</description>
 <author>Application Administrator</author>
 <pubDate>1/8/2005 10:50:30 AM</pubDate>
 </item>
 <item>
 <title>Microsoft ASP.NET Programming with Microsoft Visual C#.NET</title>
 <link>http://localhost/cms400Min/collection.aspx?id=25</link>
 <description>Teach yourself how to write high-performance Web applications with ASP.NET and
Visual C# .NET--one step at a time.</description>
 <author>Application Administrator</author>
 <pubDate>1/4/2005 7:16:21 PM</pubDate>
 </item>
 <item>
 <title>HTML for the World Wide Web with XHTML and CSS</title>
 <link>http://localhost/cms400Min/collection.aspx?id=27</link>
 <description>The easiest HTML book we've seen that still manages to be comprehensive, and to
stay firmly grounded in the reality of today's demanding users and complex browsers.</description>
 <author>Application Administrator</author>
 <pubDate>1/19/2005 4:18:06 PM</pubDate>
 </item>
 </channel>
</rss>

```

# XML Collection Function

The XML collection function wraps XML around the parameters of a list of content that's created as a collection in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the collection information. The ecmCollectionXML function allows you to easily define how the collection data appears on the Web page.

Shown here is the format of the ecmCollectionXML function.

```
<% ecmCollectionXML "id", "GetHtml" %>
```

The ecmCollectionXML function attributes are explained below.

Attribute	Description
id	The id of the collection that you would like to be displayed is defined here.
GetHtml	<p>Determines whether the HTML contained within the content is included in the XML output.</p> <p>0 = Do not get HTML 1 = Get HTML</p> <hr/> <p><i>Note: Ektron recommends using 0 (zero). HTML content can be large and this could cause display problems when your XML collection is displayed on your Web page.</i></p> <hr/>

See Also: The **Ektron CMS400.NET** User Manual Section "Working with Collections" > "Comparison of Collections, Menus, and the List Summary Features"

When using the XML Collection function, the following tags are created around the properties of each item in the collection.

XML Tags	Description
<Content></Content>	Declares a new section of content in the XML collection.
<ID></ID>	<p>The content block ID.</p> <p>For example:</p> <pre>&lt;ID&gt;27&lt;/ID&gt;</pre>

XML Tags	Description
<Title></Title>	The title of the content block. For example: <code>&lt;Title&gt;HTML for the World Wide Web with XHTML and CSS&lt;/Title&gt;</code>
<QuickLink></QuickLink>	The quicklink associated with the content block. For example: <code>&lt;QuickLink&gt;/cms400Min/collection.aspx?id=27&lt;/QuickLink&gt;</code>
<Teaser></Teaser>	The summary of the content block. For example: <code>&lt;Teaser&gt;The easiest HTML book weve seen that still manages to be comprehensive.&lt;/Teaser&gt;</code>
<Html></Html>	All of the HTML content contained in a content block. See Also: <a href="#">"GetHtml" on page 663</a> For example: <code>&lt;Html&gt;&lt;em&gt;ASP.NET Unleashed, Second Edition&lt;/em&gt; is really big, really thorough, and really &lt;i&gt;good&lt;/i&gt;. The explanations are simple and clear as anything weve seen in a .NET book. The sample code addresses virtually every facet of ASP.NET development, from little gems (code for validating credit cards) to full-fledged web stores. Best of all, Stephen Walther identifies the mistakes ASP.NET developers are most likely to make -- and presents detailed solutions. &lt;/Html&gt;</code>
<StartDate></StartDate>	The content block's start date formatted as a .NET date type For example: <code>&lt;StartDate&gt;1/1/0001 12:00:00 AM&lt;/StartDate&gt;</code>
<DateModified></DateModified>	The last date the content block was modified. For example: <code>&lt;DateModified&gt;1/19/2005 4:18:06 PM&lt;/DateModified&gt;</code>
<EndDate></EndDate>	The date the content block stops running on the Web site For example: <code>&lt;EndDate&gt;12/31/9999 12:00:00 AM&lt;/EndDate&gt;</code>

XML Tags	Description
<Title></Title>	The title of the content block. For example: <code>&lt;Title&gt;HTML for the World Wide Web with XHTML and CSS&lt;/Title&gt;</code>
<QuickLink></QuickLink>	The quicklink associated with the content block. For example: <code>&lt;QuickLink&gt;/cms400Min/collection.aspx?id=27&lt;/QuickLink&gt;</code>
<Teaser></Teaser>	The summary of the content block. For example: <code>&lt;Teaser&gt;The easiest HTML book weve seen that still manages to be comprehensive.&lt;/Teaser&gt;</code>
<Html></Html>	All of the HTML content contained in a content block. See Also: <a href="#">"GetHtml" on page 663</a> For example: <code>&lt;Html&gt;&lt;em&gt;ASP.NET Unleashed, Second Edition&lt;/em&gt; is really big, really thorough, and really &lt;i&gt;good&lt;/i&gt;. The explanations are simple and clear as anything weve seen in a .NET book. The sample code addresses virtually every facet of ASP.NET development, from little gems (code for validating credit cards) to full-fledged web stores. Best of all, Stephen Walther identifies the mistakes ASP.NET developers are most likely to make -- and presents detailed solutions. &lt;/Html&gt;</code>
<StartDate></StartDate>	The content block's start date formatted as a .NET date type For example: <code>&lt;StartDate&gt;1/1/0001 12:00:00 AM&lt;/StartDate&gt;</code>
<DateModified></DateModified>	The last date the content block was modified. For example: <code>&lt;DateModified&gt;1/19/2005 4:18:06 PM&lt;/DateModified&gt;</code>
<EndDate></EndDate>	The date the content block stops running on the Web site For example: <code>&lt;EndDate&gt;12/31/9999 12:00:00 AM&lt;/EndDate&gt;</code>

XML Tags	Description
<LastEditorFname></LastEditorFname>	The first name of the last person who edited the content block. For example: <LastEditorFname>John</LastEditorFname>
<LastEditorLname></LastEditorLname>	The last name of the last person who edited the content block. For example: <LastEditorLname>Johnson</LastEditorLname>
<Hyperlink></Hyperlink>	The hyperlink shown in the collection for linking to the content block. For example: <Hyperlink><a href="/cms400Min/collection.aspx?id=27">HTML for the World Wide Web with XHTML and CSS</a></Hyperlink>
<DisplayStartDate></DisplayStartDate>	The content block's start date. It is formatted as a string that represents Ektron CMS400's display of the date.
<FolderID></FolderID>	The ID of the folder in which the content block is contained. For example: <FolderID>0</FolderID>
<ContentStatus></ContentStatus>	The status of the content block.
<Language></Language>	The language of the content block.
<DisplayDateModified></DisplayDateModified>	edited. It is formatted as a string that represents Ektron CMS400's display of the date. For example: <DisplayDateModified>1/19/2005 4:18:06 PM</DisplayDateModified>
<DisplayEndDate></DisplayEndDate>	The content block's end date. It is formatted as a string that represents Ektron CMS400's display of the date. For example: <DisplayEndDate>12/31/9999 12:00:00 AM</DisplayEndDate>



XML Tags	Description
<code>&lt;EndDateAction&gt;&lt;/EndDateAction&gt;</code>	What should happen to the content block on its end date. <ul style="list-style-type: none"><li>• archive display</li><li>• archive expire</li><li>• refresh report</li></ul> For example: <code>&lt;EndDateAction&gt;Archive_Expire&lt;/EndDateAction&gt;</code>
<code>&lt;Comment&gt;&lt;/Comment&gt;</code>	Comments contained in the content block. For example: <code>&lt;Comment&gt;move picture below text&lt;/Comment&gt;</code>

Here is how the raw XML Collection output appears before an XSLT is applied.

```

<Content>
 <ID>24</ID>
 <Title>ASP.NET Unleashed</Title>
 <QuickLink>/CMS400Min/collection.aspx?id=24</QuickLink>
 <Teaser><p>ASP.NET Unleashed, Second Edition is really big, really thorough, and really <i>good</i>.</p></Teaser>
 <Html></Html>
 <StartDate>1/1/0001 12:00:00 AM</StartDate>
 <DateModified>1/12/2005 7:53:31 PM</DateModified>
 <EndDate>12/31/9999 12:00:00 AM</EndDate>
 <LastEditorFname>Application</LastEditorFname>
 <LastEditorLname>Administrator</LastEditorLname>
 <Hyperlink>ASP.NET Unleashed</Hyperlink>
 <DisplayStartDate></DisplayStartDate>
 <FolderID>0</FolderID>
 <ContentStatus></ContentStatus>
 <Language>0</Language>
 <DisplayDateModified>1/12/2005 7:53:31 PM</DisplayDateModified>
 <DisplayEndDate></DisplayEndDate>
 <EndDateAction>Archive_Expire</EndDateAction>
 <Comment></Comment>
</Content>
<Content>
 <ID>25</ID>
 <Title>Microsoft ASP.NET Programming with Microsoft Visual C#.NET</Title>
 <QuickLink>/CMS400Min/collection.aspx?id=25</QuickLink>
 <Teaser><p>Teach yourself how to write high-performance Web applications with ASP.NET and Visual C# .NET--one step at a time.</p></Teaser>
 <Html></Html>
 <StartDate>1/1/0001 12:00:00 AM</StartDate>
 <DateModified>1/4/2005 7:16:21 PM</DateModified>
 <EndDate>12/31/9999 12:00:00 AM</EndDate>
 <LastEditorFname>Application</LastEditorFname>
 <LastEditorLname>Administrator</LastEditorLname>
 <Hyperlink>Microsoft ASP.NET Programming with Microsoft Visual C#.NET</Hyperlink>
 <DisplayStartDate></DisplayStartDate>
 <FolderID>0</FolderID>
 <ContentStatus></ContentStatus>
 <Language>0</Language>
 <DisplayDateModified>1/4/2005 7:16:21 PM</DisplayDateModified>
 <DisplayEndDate></DisplayEndDate>
 <EndDateAction>Archive_Expire</EndDateAction>
 <Comment></Comment>
</Content>

```

# Random Content

The random content function is used to randomly display the content of a content block that belongs in a specified collection.

Displayed here is the format for the random content function

```
<% ShowRandomContent "id", XSLTid %>
```

Where the "id" represents the ID number of the collection the content blocks, that will be displayed, belong to.

View Collection "Products"

Title

RC Sportster

RC Cheetah

RC Lilly

RC Redstar

[More info](#)

Title:	Products
ID:	1
Template:	/CMS8888/sample/products.asp
Last User To Edit:	Application Administrator
Last Edit Date:	29-Oct-2002 09:36 PM
Date Created:	29-Oct-2002 09:34 PM
...	...

And "XSLTid" is the ID number of the display XSLT that will be applied to the content blocks in the collections.

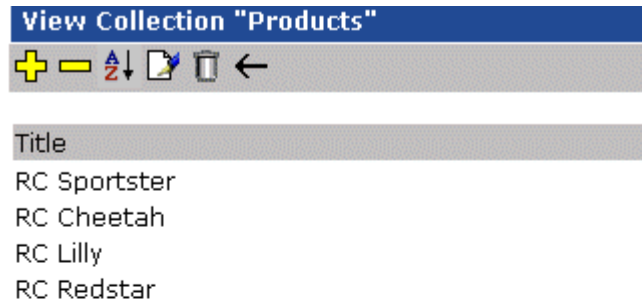
# Random Summary

Similar to the random content function, the random summary function randomly displays a content block summary that belongs in a specified collection.

Here is the format for the random content function

```
<% ShowRandomSummary "id" %>
```

The "id" represents ID number of the collection to which the content blocks belong.



[More info](#)



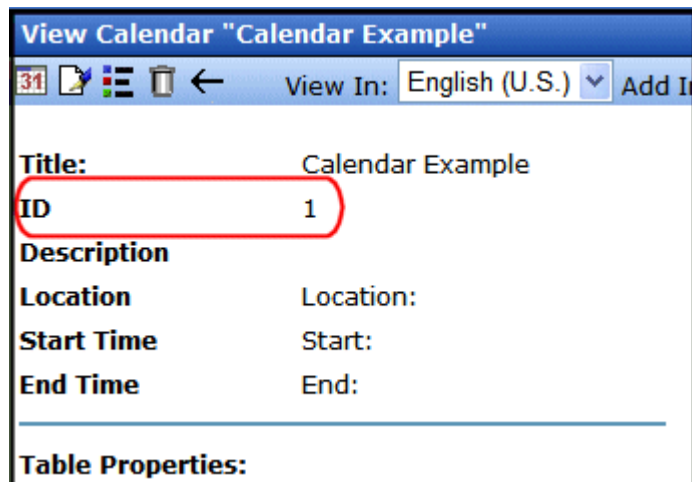
# Event Calendar Function

The ecmEvtCalendar function displays a month view event calendar that has been created in Ektron CMS400.NET. Here is the format for the event calendar custom function.

```
<% ecmEvtCalendar "id" %>
```

**NOTE** You can use ecmCalendar to call a calendar and display a different view, such as daily. For more information, see "Calendar Function" on page 674

The "id" argument represents the ID number of the event calendar in the Workarea.



## Event Calendar Style Sheet



































































Before you insert an event Calendar into your Ektron CMS400.NET template, you should make a reference to the style sheet, `\siteroot\Workarea\csslib\calendarStyles.css`, in your template. This style sheet contains styles that define the Calendar.

## Event Calendar Sample

Here is an example of an event Calendar in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <% ecmEvtCalendar "1" %>
 </td>
</tr>
```

When this code is displayed in a browser, the Calendar with an ID-1 is displayed.  
Here is the calendar displayed on a Web Page

If...	Calendar Display																																																							
Logged in	<div><div>&lt;&lt; May</div><div>June, 2005</div><div>July</div><table><tr><th>Monday</th><th>Tuesday</th><th>Wednesday</th><th>Thursday</th><th>Friday</th></tr><tr><td></td><td></td><td> 1</td><td> 2</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td> 6</td><td> 7</td><td> 8 Centaur's Online Marketing 2005</td><td> 9 Centaur's Online Marketing 2005</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td> 13</td><td> 14 Internet World</td><td> 15</td><td> 16</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td> 20</td><td> 21</td><td> 22</td><td> 23</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td> 27</td><td> 28</td><td> 29</td><td> 30</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table></div>	Monday	Tuesday	Wednesday	Thursday	Friday			 1	 2							 6	 7	 8 Centaur's Online Marketing 2005	 9 Centaur's Online Marketing 2005							 13	 14 Internet World	 15	 16							 20	 21	 22	 23							 27	 28	 29	 30						
Monday	Tuesday	Wednesday	Thursday	Friday																																																				
		 1	 2																																																					
 6	 7	 8 Centaur's Online Marketing 2005	 9 Centaur's Online Marketing 2005																																																					
 13	 14 Internet World	 15	 16																																																					
 20	 21	 22	 23																																																					
 27	 28	 29	 30																																																					

If...	Calendar Display																																																																																																																																																																																																																																																																
Logged Out	<table><tr><th colspan="12">&lt; May</th><th colspan="12">June, 2005</th><th colspan="12">July &gt;&gt;</th></tr><tr><th colspan="4">Monday</th><th colspan="4">Tuesday</th><th colspan="4">Wednesday</th><th colspan="4">Thursday</th><th colspan="4">Friday</th></tr><tr><td colspan="4"></td><td colspan="4"></td><td colspan="4">1</td><td colspan="4">2</td><td colspan="4">3</td></tr><tr><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td></tr><tr><td colspan="4">6</td><td colspan="4">7</td><td colspan="4">8</td><td colspan="4">9</td><td colspan="4">10</td></tr><tr><td colspan="4"></td><td colspan="4"></td><td colspan="4">Centaur's Online Marketing 2005</td><td colspan="4">Centaur's Online Marketing 2005</td><td colspan="4"></td></tr><tr><td colspan="4">13</td><td colspan="4">14</td><td colspan="4">15</td><td colspan="4">16</td><td colspan="4">17</td></tr><tr><td colspan="4"></td><td colspan="4">Internet World</td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td></tr><tr><td colspan="4">20</td><td colspan="4">21</td><td colspan="4">22</td><td colspan="4">23</td><td colspan="4">24</td></tr><tr><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td></tr><tr><td colspan="4">27</td><td colspan="4">28</td><td colspan="4">29</td><td colspan="4">30</td><td colspan="4"></td></tr><tr><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td><td colspan="4"></td></tr></table>	< May												June, 2005												July >>												Monday				Tuesday				Wednesday				Thursday				Friday												1				2				3																								6				7				8				9				10												Centaur's Online Marketing 2005				Centaur's Online Marketing 2005								13				14				15				16				17								Internet World																20				21				22				23				24																								27				28				29				30																											
< May												June, 2005												July >>																																																																																																																																																																																																																																									
Monday				Tuesday				Wednesday				Thursday				Friday																																																																																																																																																																																																																																																	
								1				2				3																																																																																																																																																																																																																																																	
6				7				8				9				10																																																																																																																																																																																																																																																	
								Centaur's Online Marketing 2005				Centaur's Online Marketing 2005																																																																																																																																																																																																																																																					
13				14				15				16				17																																																																																																																																																																																																																																																	
				Internet World																																																																																																																																																																																																																																																													
20				21				22				23				24																																																																																																																																																																																																																																																	
27				28				29				30																																																																																																																																																																																																																																																					

As you can see, when you are logged into the Web site, you can add and view Calendar events. When logged out, you can only view calendar events.

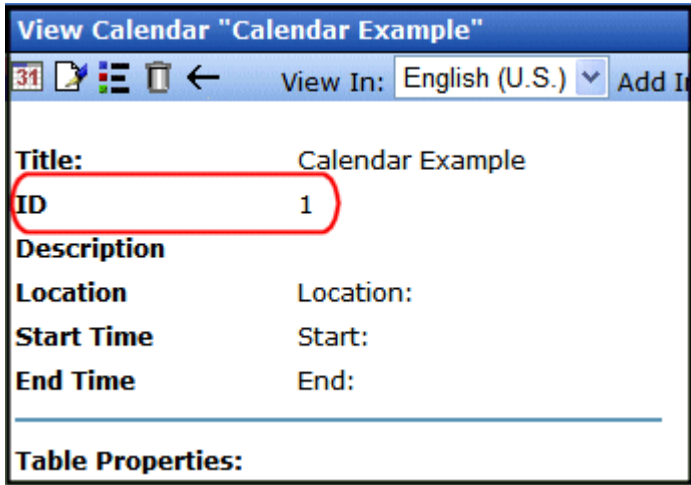
# Calendar Function

The ecmCalendar function displays a calendar that has been created in Ektron CMS400.NET. Here is the format for the event calendar custom function.

```
<% ecmCalendar "id", "DisplayType" %>
```

**NOTE** While you can display a month view calendar with ecmCalendar, you can also use ecmEvtCalendar to display a month view calendar. For more information, see ["Event Calendar Function" on page 671](#)

The ecmCalendar function attributes are explained below.

Attribute	Description
id	<p>The "id" argument represents the ID number of the event calendar in the Workarea.</p> 



Attribute	Description																																																								
DisplayType	<p>Specify the Calendar display type.</p> <ul style="list-style-type: none"><li><b>Month</b> - A 30 day calendar month view.</li></ul> <div><div>&lt;&lt; November</div><div>December 2005</div><div>January &gt;&gt;</div><table><tr><th>Monday</th><th>Tuesday</th><th>Wednesday</th><th>Thursday</th><th>Friday</th></tr><tr><td></td><td></td><td></td><td>1</td><td>2</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr></table></div> <ul style="list-style-type: none"><li><b>Day</b> - Single day view.</li></ul> <div><div>&lt;&lt;</div><div>12/14/2005</div><div>Thursday, December 15, 2005</div><div>12/16/2005 &gt;&gt;</div><table><tr><td>12:00 AM</td><td></td></tr><tr><td>1:00 AM</td><td></td></tr><tr><td>2:00 AM</td><td></td></tr></table></div> <ul style="list-style-type: none"><li><b>Inline</b> - Display events that occur from the start date of the calendar through the end date.</li></ul> <div><div>Thursday, December 01, 2005 - Saturday, December 31, 2005</div><div>12/15/2005</div><div>Monthly Audit Business Office Start:8:00 AM</div></div> <ul style="list-style-type: none"><li><b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.</li></ul> <div><div>&lt;&lt;</div><div>December, Jan</div><div>Hov 2005 &gt;&gt;</div><div><div>Mon</div><div>Tue</div><div>Wed</div><div>Thu</div><div>Fri</div></div><table><tr><td></td><td></td><td></td><td>1</td><td>2</td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr><tr><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td></tr><tr><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td></tr></table><div><div><div><div></div><div></div></div><div>12/21/2005</div><div>Christmas Party</div><div>Location:Event Room</div><div>Start:7:30 PM</div><div>End:11:00 PM</div><div>Come to the Christmas party!</div></div></div></div>	Monday	Tuesday	Wednesday	Thursday	Friday				1	2																5	6	7	8	9	12:00 AM		1:00 AM		2:00 AM					1	2	5	6	7	8	9	12	13	14	15	16	19	20	21	22	23
Monday	Tuesday	Wednesday	Thursday	Friday																																																					
			1	2																																																					
5	6	7	8	9																																																					
12:00 AM																																																									
1:00 AM																																																									
2:00 AM																																																									
			1	2																																																					
5	6	7	8	9																																																					
12	13	14	15	16																																																					
19	20	21	22	23																																																					

## Calendar Style Sheet

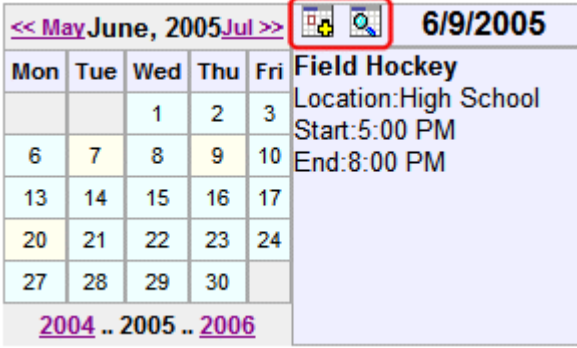
Before you insert an event Calendar into your Ektron CMS400.NET template, you should make a reference to the style sheet, `\siteroot\Workarea\csslib\calendarStyles.css`, in your template. This style sheet contains styles that define the Calendar.

## Event Calendar Sample

Here is an example of an event Calendar in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <% ecmCalendar "1", "MonthUpWithEvent" %>
 </td>
</tr>
```

When this code is displayed in a browser, the Calendar with an ID of 1 and DisplayType of MonthUpWithEvent is displayed. Here is the calendar displayed on a Web Page

If...	Calendar Display
Logged in	<p>Notice the add event and view event buttons that appear when logged in.</p> 

If...	Calendar Display																																																
Logged Out	<table><tr><td colspan="5">&lt;&lt; MayJune, 2005Jul &gt;&gt;</td><td>6/9/2005</td></tr><tr><td>Mon</td><td>Tue</td><td>Wed</td><td>Thu</td><td>Fri</td><td>Field Hockey</td></tr><tr><td></td><td></td><td>1</td><td>2</td><td>3</td><td>Location:High School</td></tr><tr><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>Start:5:00 PM</td></tr><tr><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>End:8:00 PM</td></tr><tr><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td></td></tr><tr><td>27</td><td>28</td><td>29</td><td>30</td><td></td><td></td></tr><tr><td colspan="5">2004 .. 2005 .. 2006</td><td></td></tr></table>	<< MayJune, 2005Jul >>					6/9/2005	Mon	Tue	Wed	Thu	Fri	Field Hockey			1	2	3	Location:High School	6	7	8	9	10	Start:5:00 PM	13	14	15	16	17	End:8:00 PM	20	21	22	23	24		27	28	29	30			2004 .. 2005 .. 2006					
<< MayJune, 2005Jul >>					6/9/2005																																												
Mon	Tue	Wed	Thu	Fri	Field Hockey																																												
		1	2	3	Location:High School																																												
6	7	8	9	10	Start:5:00 PM																																												
13	14	15	16	17	End:8:00 PM																																												
20	21	22	23	24																																													
27	28	29	30																																														
2004 .. 2005 .. 2006																																																	

As you can see, when you are logged into the Web site, you can add and view Calendar events. When logged out, you can only view calendar events.

# Form Function

The `ecmFormBlock` and `ecmFormBlockNoFormTag` functions display an HTML form content block. Both functions have the same parameters and display the form the same way. The only difference is `ecmFormBlock` automatically inserts form tags, `ecmFormBlockNoFormTag` does not.

The following is an example of form tags.

```
<form name="myform" action="#" onsubmit="return
EkFmValidate(this);" method="post">
```

```
</form>
```

When using `ecmFormBlock`, form tags are added automatically. When using `ecmFormBlockNoFormTag`, form tags need to be added manually.

For more information about HTML forms, see the **Ektron CMS400.NET** Administrator manual chapter “Working with HTML Forms.”

---

**IMPORTANT!**

If you create a template for an existing form content block, you must manually change its quicklink to point to the new template. This change does not occur automatically. This procedure is described in the Ektron CMS400.NET Administrator manual chapter “Managing Library Assets”, section “Updating Default Template for Multiple Quicklinks.”

---

Here are the two formats for the form custom function.

```
<% ecmFormBlock “id” %>
or
<% ecmFormBlockNoFormTag “id” %>
```

---

**NOTE**

As an alternative, you can use the multipurpose function, which can display either a content block or a content block associated with a form. For more information, see “MultiPurpose Function” on page 680.

---

The "id" argument represents the form's ID number, visible when viewing the form from the View Form screen.

ecmFormBlock and ecmFormBlockNoFormTag are interchangeable when using this function. Here is an example of a form content block in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <% ecmFormBlock "367" %>
 </td>
</tr>
```

When this code is read by a browser, the following is displayed.

When a visitor to your site submits a form, the form uses the form information to process the data.

# MultiPurpose Function

The ecmMultiPurpose function displays either a content block or an content block associated with a form. As a result, you can use one template to display both types of content block.

Here is the format for the Multipurpose function:

`<% ecmMultiPurpose "id" %>`

To pass in a standard content block, pass the id value in the url parameter, as illustrated below.

`http://localhost/cms400sample/index.asp?id=2`

To pass in a Form content block, pass the ekfrm value in the url parameter, as illustrated below.

`http://localhost/cms400sample/index.asp?ekfrm=2`

---

**NOTE** For more information about passing the id value as a URL parameter, see ["Dynamic Content Block" on page 612](#).

---

If the id value or ekfrm value does not exist, a blank template appears.

If the url parameter is not passed in or recognized, the function calls ecmContentBlock and passes the DefaultID value as the id argument. If the DefaultID value is zero (0), ecmContentBlock is not called.

# Metadata Function

The metadata tag is used to display the metadata for one or more content blocks in the Ektron CMS400.NET Web site. This custom tag lets you generate the list of metadata edited by the authors and insert it in the source of the Web page. Below is an example of the metadata custom tag.

```
<% ecmMetaData "ItemList", "Spare1" %>
```

The attributes are described below.

Attribute	Description	Required?
ItemList	List of the content block ID numbers, and exclusions See Also: <a href="#">"Creating an Item List" on page 681</a> and <a href="#">"Item List" on page 684</a>	No
Spare1	Not yet implemented See Also: <a href="#">"Spare 1" on page 686</a>	MUST BE ""

## Creating an Item List

The Item List for the ecmMetadata tag has the following format.

```
"[content block id; Exclusion list]...[content block id; Exclusion list]"
```

Attribute	Description	Required?
[ (left bracket)	Delineates the beginning of a content block id/exclusion list pair	Yes
Content Block ID	The ID of the content block whose metadata you want to retrieve	Yes
; (semicolon)	Separator: Separates content block id from the exclusion list	No (Yes if including an exclusion list)
Exclusion List	A comma delimited list of metadata you do not want to be retrieved for this content block.	No
] (right bracket)	Delineates the end of a content block id/exclusion list pair	Yes

## ecmMetadata Example

For this example, our administrator has defined the following metadata types for Ektron CMS400.NET.

Name	Style	Type	Editable	Required	Remove Dups	Case Sensitive
Title	n/a	HTML	Yes	Yes	No	n/a
Keywords	Name	Meta	Yes	Yes	Yes	No
Content-Type	http-equiv	Meta	No	Yes	Yes	n/a

Our example Web page has two content blocks displayed.

- content block 1, the company's home page introduction text
- content block 2, a sidebar containing the lead story for the week

The metadata entered by your content editors for these two content blocks are:

### Content Block 1:

Title – "Welcome to Our Home Page"

Keywords – "Home; page; company; Welcome"

Content-Type -- "text/html; charset=iso-8859-1"

### Content Block 2:

Title – "Announcing our New Product!"

Keywords – "Announce; company; product; widget"

Content-Type -- "text/html; charset=iso-8859-1"

To include both content blocks' metadata in a Web page, the ecmMetadata call looks like this.

```
<html>
 <head>
 <% ecmMetadata "[1][2;Title,Content-Type]", "" %>
 (rest of head)
 </head>
 <body>
 (rest of web page document)
 </body>
</html>
```

In the above example, the ecmMetadata call is contained in the <head> of the HTML document (since all HTML metadata is required to exist there). When the Web page is viewed, the Web page source contains the following information:

```
<html>
 <head>
 <Title>Welcome to Our Home Page</Title>
 <meta name="Keywords" content= Home; page; company; Welcome; Announce;
 product; widget">
 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```



```

 (rest of head)
</head>
<body>
 (rest of web page document)
</body>
</html>

```

The ecmMetadata constructed three HTML tags based on the administrator defined parameters and the ecmMetadata call.

The metadata tag was constructed in the page because:

- **[1]** - Displays all metadata content for the content block ID=1. In this example, the title, keywords, and content-type were defined, so they were displayed as:

```

<Title>Welcome to Our Home Page</Title>
<meta name="Keywords" content= Home; page; company; Welcome;
 Announce; product; widget">
<meta http-equiv="Content-Type" content="text/html;
 charset=iso-8859-1">

```

- **[2;title,content-type]** - Displays all metadata content for content block ID=2, except for its title and content-type definitions.

```

<Title>Welcome to Our Home Page</Title>
<meta name="Keywords" content= Home; page; company; Welcome;
 Announce; product; widget">
<meta http-equiv="Content-Type" content="text/html;
 charset=iso-8859-1">

```

The next section provides more examples.

## More ecmMetadata Examples

Shown below is an example of the metadata function in an ASP Web page.

```

<TR>
 <TD>
 <% ecmMetaData "[1][2;title][14;keywords]", "" %>
 </TD>
</TR>

```

The previous example has the following properties:

- All metadata for content block ID=1 is inserted into the page
- All metadata for the content block ID=2 except for its title is inserted into the page
- All metadata for the content block ID=14 except for its keywords is inserted into the page

## Using a Dynamic Metadata Tag

You can use a metadata tag in a template to define the metadata for a specific content block. However, you can also use a dynamic metadata tag combined with a content block tag to show the metadata for all content blocks on a Web page.

This allows you to create a template and not have to keep updating the tag by adding the new ID and exclusion list. Shown below is the sample code for this.

```
<!-- #include virtual="/CMS400.NETsample/CMS400.NETscripts/ektronAPI.asp" -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
 <head>
 <% Dim id
 if (request.QueryString("id")<> "") then
 id = request.QueryString("id")
 else
 id = 1
 end if
 ecmMetadata "[" & id & "]", ""
 %>
 </head>
 <body>
 <% ecmContentBlock(request.QueryString("id")) %>
 </body>
</html>
```

In this example, if you specify a content block in the URL and the page is displayed, the metadata takes that ID and displays the proper metadata in the source for the content block.

## ecmMetadata Attributes

Below are the attributes and their descriptions.

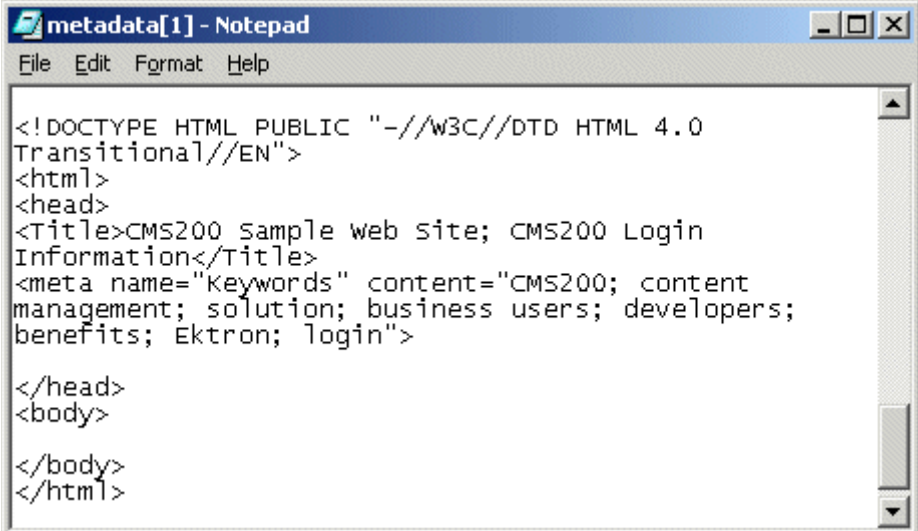
### Item List

In this attribute of the ecmMetadata tag, you need to specify two values.

Value	Specifies
ID	The content block ID whose metadata will appear
Exclusion List	The metadata to be excluded. See Also: <a href="#">"Creating an Item List" on page 681</a>

Some examples might be:

```
<% ecmMetaData "[1][2]", "" %>
```



```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 sample web site; CMS200 Login
Information</Title>
<meta name="keywords" content="CMS200; content
management; solution; business users; developers;
benefits; Ektron; login">

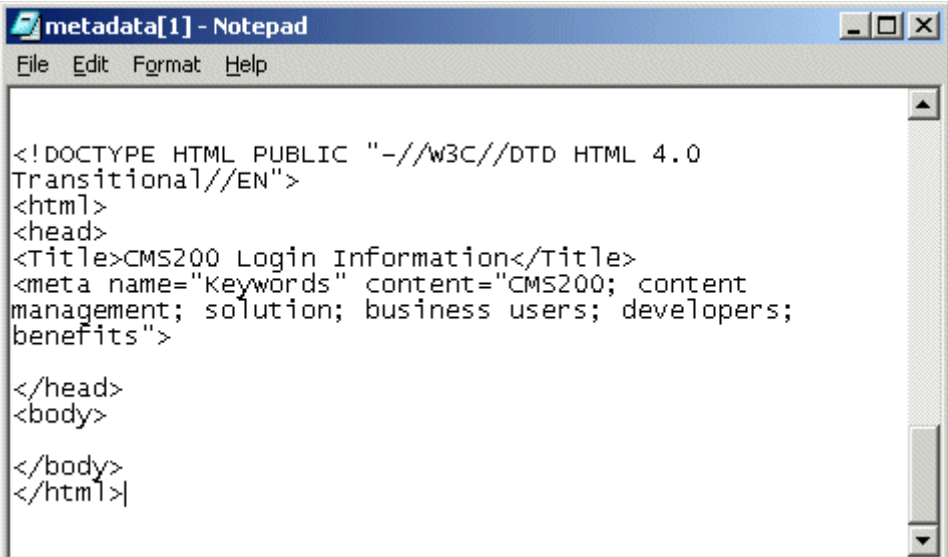
</head>
<body>

</body>
</html>

```

In this example, all metadata from content block ID=1 and 2 is displayed.

```
<% ecmMetaData "[1;title][2;keywords]", "" %>
```



```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 Login Information</Title>
<meta name="keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

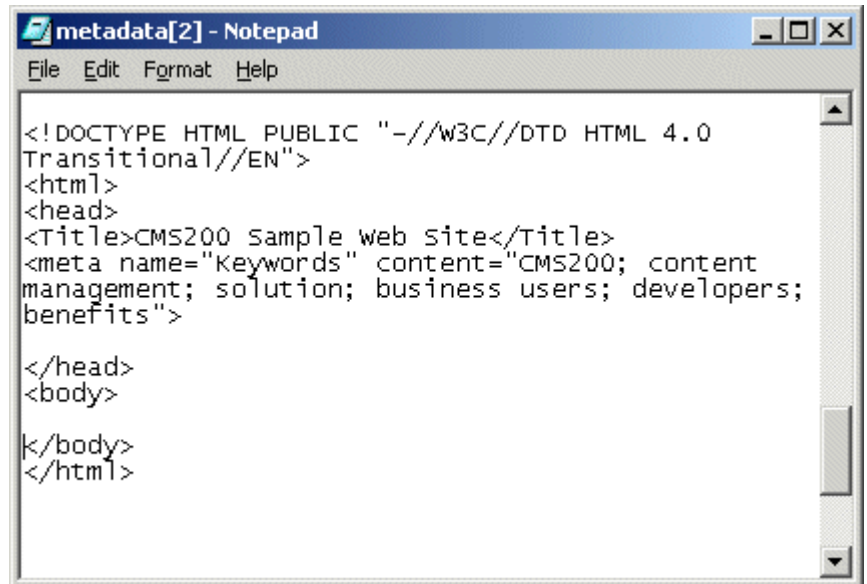
</head>
<body>

</body>
</html>

```

In this example, all the metadata from content block ID=1 will be displayed *except* for the title. Also, all the metadata for content block ID=2 is displayed *except* its keywords.

```
<% ecmMetaData "[1][2:title,keywords]","", "" %>
```



```
metadata[2] - Notepad
File Edit Format Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<title>CMS200 Sample web site</Title>
<meta name="keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

k/body>
</html>
```

In this example, all metadata from content block ID=1 is displayed, and all metadata from content block ID=2 is displayed *except* its title and keywords.

## Spare 1

One attribute field at the end is currently not used. It is for future scalability. When you insert a single metadata tag into a template, you must leave this attributes as "".

# Show Site Language Function

This function lets a visitor to your Web site select a language in which to view the site. Here is what the function looks like when published on a Web page.



It lists all languages selected on the **Settings > Configuration > Language settings** screen. (For more information, see the “Multi-Language Support” chapter of the **Ektron CMS400.NET** Administrator manual.)

Below is an example of the Show Site Language custom tag.

```
<% ecmShowSiteLanguage(Fieldname)%>
```

You can place this function in any location of any page on your site. By default, it appears in the lower right corner of the home page.

**NOTE** This function only works with Asp and ASP.Net.

The attribute is described in this table.

Attribute	Description
Fieldname	<p>An optional argument. You can leave it blank and allow the drop-down list to be populated by the available languages for your Web site. For example:</p> <pre>&lt;%=ecmShowSiteLanguage('')%&gt;</pre> <p>Or you can pass in an argument such as:</p> <pre>&lt;%=ecmShowSiteLanguage('mysitelanguage')%&gt;</pre>

# DHTML Menus

**Ektron CMS400.NET**'s Menu feature lets users create and maintain a dropdown menu system for your Web site. The Menu feature lets you display menu options that link to content blocks, library assets, external hyperlinks and submenus.

To learn more about menus, please refer to the "Menu Feature" chapter of the **Ektron CMS400.NET** User Manual.

## Additional Documentation in Ektron Knowledge Base

This chapter explains how to implement standard menus that ship with **Ektron CMS400.NET**. Ektron's Web site also provides Knowledge Base articles that explain how to extend the standard menu features.

## Displaying a Menu on a Web Page

### GenerateMenu and DropMenu

---

**IMPORTANT!**

The menu functions described below are only one example of how to implement menus. The scripts reside in the Application API file. You can implement your own menu system using JavaScript.

---

Use the `GenerateNextGenMenu` and `DropNextGenMenu` custom functions to display a DHTML menu on a Web page. Place `GenerateNextGenMenu` once on the page to provide an infrastructure for `DropNextGenMenu`. Then, place `DropNextGenMenu` as needed to implement individual menus.

Follow these steps to display a menu on a Web template.

1. Place the following code directly after the page's `<body>` tag.

```
<%= (ecmDHTML_GenerateNextGenMenu("menu id number")) %>
```

For example:

```
<%= (ecmDHTML_GenerateNextGenMenu("1")) %>
```

To display more than one menu, separate each id number with a comma. For example:

```
<%= (ecmDHTML_GenerateNextGenMenu("1, 2, 3")) %>
```

2. If the user clicks *text* to display the menu, place the following code within the template file where the text would appear.

```
<%= ecmDHTML_DropNextGenMenu(menu id number,"text user clicks to display menu")%>
```

For example:

```
<%= ecmDHTML_DropNextGenMenu(1,"Products")%>
```

If the user clicks a *graphic* to invoke the menu, use this syntax:

```
<%= ecmDHTML_DropNextGenMenu(menu id number,'')%>
```

For example:

```
<%= ecmDHTML_DropNextGenMenu(1,'')%>
```

## Displaying Menus in a Multi-language System

If your Web site supports multiple languages, and the user selects a language before viewing the menu, if an edition of the menu is available in that language, it appears. If not, nothing appears.

For more information, see the **Ektron CMS400.NET** Administrator Manual section “Multi-Language Support” > “Working with Multi-Language Content” > “Working with Menus in a Multi-Language System.”

## Customizing the Menu's Appearance

You can customize a menu's appearance, such as the font style, color, and size. So, you have the flexibility to coordinate a menu with the design your Web site.

To customize a menu in **Ektron CMS400.NET**, you can change the `ek_menu.js` file. See Also: ["ek\\_Menu.js" on page 689](#)

## ek\_Menu.js


You can change the look and feel of a menu by changing the `ek_menu.js` file. If you used the default settings when installing **Ektron CMS400.NET**, this file is in `webroot/siteroot/Workarea/java/`.

Use the following table to help you change menu attributes.

Attribute name	Specifies the	Default value
fontSize	Size of menu text	14
fontWeight	Thickness of menu text You can set FontWeight by name (Bold or Plain) or value. Values range from 0 -1000.	Plain

Attribute name	Specifies the	Default value
fontFamily	Font style of menu text <i>See Also: "FontFamily" on page 691</i>	arial,helvetica,espy, sans-serif
fontColor	Color of menu text  <u>Note: When changing the color, you can specify the color name or code.</u>	#000000
fontColorHiLite	Color of text when menu item is highlighted	#ffffff
bgColor	Color of border between menu items	#555555
menuBorder	Width (in pixels) of border around main menu	1
menuItemBorder	Width (in pixels) of border around each menu item	1
menuItemBgColor	Background color of menu items	#6699ff
menuLiteBgColor	Color of main menu border highlight <i>See Also: "menuLiteBgColor" on page 691</i>	#ffffff
menuBorderBgColor	Color of the main menu border	#6699ff
menuHiLiteBgColor	Background color of highlighted menu item	#000084
menuContainerBgColor	Background color of the <i>container</i>  <u>Note: The container is like a &lt;div&gt; tag that displays the menu. When you set the container color, it appears at the top and left side of the menu frame only.</u>	#6699ff



Attribute name	Specifies the	Default value
childMenuIcon	<p>Path to the submenu icon (the image used to indicate that a submenu is available from this menu item).</p>  <p>To display a different image, either provide an external path to it or upload the image to <b>Ektron CMS400.NET</b> then provide a local path.</p>	./Workarea/ images/ application/ arrows.gif
childMenuIconHiLite	<p>The path to the icon that appears when you hover the mouse over a menu item that leads to a submenu.</p> <p>To display a different image, either provide an external path to it or upload the image to <b>Ektron CMS400.NET</b> then provide a local path.</p>	./Workarea/ images/ application/ arrows2.gif

## FontFamily

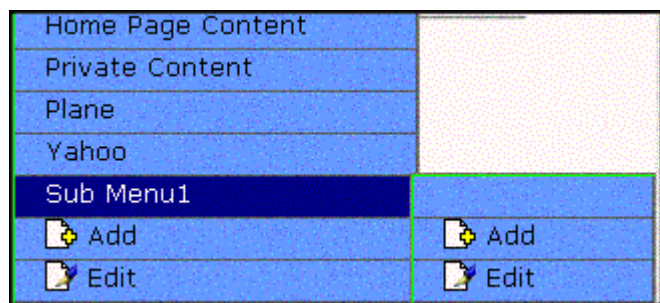
Use the fontFamily attribute to specify the menu font. It must be supported by the browser.

You can assign to this property several, comma-separated values.

**NOTE** To display a font family other than the browser default, specify only the font name and no other comma-separated values. If you specify more than one value for the font family, Ektron CMS400.NET uses the default IE settings.

## menuLiteBgColor

### Example



# XML Menu

The ecmGetMenuXML function wraps XML around the properties of a menu created in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the menu. The ecmGetMenuXML function allows you to easily define how the menu data appears on the Web page.

Shown here is the format of the ecmGetMenuXML function.

```
<% ecmGetMenuXML (id) %>
```

The ecmGetMenuXml function attributes are explained below.

Attribute	Description
Id	The id that associated with the menu.

Here is how the raw XML Menu output appears before an XSLT is applied.

```
<Item>
 <ItemID>1</ItemID>
 <ItemType>Submenu</ItemType>
 <ItemTitle>MenuExample</ItemTitle>
 <ItemImage></ItemImage>
 <ItemImageOverride>False</ItemImageOverride>
 <Menu>
 <ID>1</ID>
 <Title>MenuExample</Title>
 <Template></Template>
 <Type>content</Type>
 <Link></Link>
 <ParentID>0</ParentID>
 <AncestorID>1</AncestorID>
 <FolderID>0</FolderID>
 <Description></Description>
 <ImageOverride>False</ImageOverride>
 </Item>
 <Item>
 <ItemID>2</ItemID>
 <ItemType>Submenu</ItemType>
 <ItemTitle>Book</ItemTitle>
 <ItemImage></ItemImage>
 <ItemImageOverride>False</ItemImageOverride>
 <Menu>
 <ID>2</ID>
 <Title>Book</Title>
 <Template></Template>
 <Type>content</Type>
 <ParentID>1</ParentID>
 <ImageOverride>False</ImageOverride>
 <Item>
 <ItemID>23</ItemID>
 <ItemType>content</ItemType>
 <ItemTitle>VB.NET How to</ItemTitle>
 <ItemImage></ItemImage>
 <ItemLink>/CMS400Min/collection.aspx?id=23</ItemLink>
 <ItemTarget>2</ItemTarget>
 </Item>
 </Menu>
</Item>
```

# Custom ColdFusion Functions

Ektron CMS400.NET maintains and display content through the use of custom Ektron ColdFusion functions and ComObjects.

When using custom ColdFusion functions, the following paths must be updated in the application.cfm.

```
<cfset application.WSDLpath = "http://localhost/siteroot/WorkArea/WebServices/ContentWS.asmx?WSDL" />
```

and

```
<cfset application.AppPath = "/siteroot/WorkArea/platforms/ColdFusion/" />
```

The application.WSDLpath needs to be updated with the location of the WSDL file. For example, if your site is [www.example.com](http://www.example.com), the WSDL path would be <http://www.example.com/WorkArea/WebServices/ContentWS.asmx>.

The application.AppPath needs to be updated with the path to the ColdFusion custom Functions.

To customize your Ektron CMS400.NET application, you will need to be familiar with these tags

Function	Description	More Information
<b>Calendar Functions</b>		
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 706
<b>Collection Functions</b>		
Collection	Displays links generated by collections	"Collection Function" on page 702
CollectionXML	Creates XML for a collection by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML Collection Function" on page 706
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 698
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 699
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 706

Function	Description	More Information
<b>Content Functions</b>		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> programmatically.	"Add Content" on page 697
Content Block	Displays and manages a content block	"Content Block Function" on page 696
Form	Displays a content block associated with a form	"Form Function" on page 709
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 709
<b>Language Functions</b>		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 713
<b>List Summary Functions</b>		
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 702
List Summary	Displays a list of content block summaries	"List Summary Function" on page 700
List Summary XML	Creates XML for a list summary by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML List Summary Function" on page 701
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 701
Single Summary	Displays the summary for a content block.	"Single Summary Function" on page 702
<b>Login Functions</b>		
Login	Adds the login or logout button	"Login Function" on page 696
<b>Menu Functions</b>		

Function	Description	More Information
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 711
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 711
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 713
<b>Metadata Functions</b>		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 710
<b>Search Functions</b>		
Search	Inserts a text box with a search button	"Search Function" on page 699
Search Display	Creates an area to display search results	"Search Display Function" on page 700

## Login Function

This tag is responsible for displaying the following buttons on the Web page.

This function is basically the same as the Login function in ASP. See "[Login Function](#)" on page 607 for details. The only difference is the syntax used to implement the function. This is listed below.

### Login Tag Example

```
<tr>
 <td>
 <cfmodule template="#request.ecm.AppPath#ecmLogin.cfm">
 </td>
</tr>
```

## Content Block Function

The content block and content blockEX tags are used to display a content block on an Ektron CMS400.NET Web page.

Shown here is the format for the content block function:

```
<CFMODULE template="#request.ecm.AppPath#ecmContentBlock.cfm"
 ID= " "
>
```

Shown here is the format for the content blockEX function:

```
<CFMODULE template="#request.ecm.AppPath#ecmContentBlockEx.cfm"
 ID=""
 XSLT=""
 Override=""
>
```

This functions are basically the same as the ContentBlock and ContentBlockEX function in ASP. See ["Content Block Function" on page 609](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion examples are below.

## Static Content Block

```
<tr>
 <td>
 <CFMODULE id="1" template="#request.ecm.AppPath#ecmContentBlock.cfm">
 </td>
</tr>
```

## Dynamic Content Block

```
<tr>
 <td>
 <CFMODULE id="#id#" template="#request.ecm.AppPath#ecmContentBlock.cfm">
 </td>
</tr>
```

## XML Content Block

```
<tr>
 <td>
 <CFMODULE id="13" XSLT="" override="1"
 template="#request.ecm.AppPath#ecmContentBlockEx.cfm">
 </td>
</tr>
```

## Add Content

The add content function allows you to programmatically add a new content block to Ektron CMS400.NET without the need of logging in.

Here is the format for the add content function:

```
<CFModule template="#request.ecm.AppPath#ecmAddContent.cfm" Username="",
Password="",
Domain="",
content_title="",
content_comment="",
ContentHtml="",
SummaryHTML="",
ContentLanguage="",
FolderID="",
GoLive="",
EndDate="",
MetaInfo="" >
```

This function is basically the same as the Add Content function in ASP. See ["Add Content" on page 617](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

## Add Content Example

The following is an example of the add content function in a template:

```
<tr>
 <td>
 <CFModule template="#request.ecm.AppPath#ecmAddContent.cfm "AddContent
 UserName="admin", Password="admin", Domain="", content_title="Content Title",
 content_comment="Comment", ContentHTML="This is the body of the content.",
 SummaryHTML="This is the <i>summary</i> for the content.", ContentLanguage="1033",
 FolderID="0", GoLive="", EndDate="", MetaInfo="">
 </td>
</tr>
```

## Random Content Function

The random content function is used to randomly display the content of a content block that belongs to a specified collection. Below is the format for the random content function:

```
<CFMODULE template="#request.ecm.AppPath#ecmShowRandomContent.cfm"
 ID=" "
 XSLT=" "
 Override=" "
>
```

This function is basically the same as the Random Content function in ASP. See ["Random Content" on page 669](#) for details.



## Random Summary Function

Similar to the random content function, the random summary function can randomly display a content block summary that belongs to a specified collection. Here is the format for the random content function

```
<CFMODULE template="#request.ecm.AppPath#ecmShowRandomSummary.cfm"
 ID=" "
>
```

This function is basically the same as the Random Summary function in ASP. See ["Random Summary" on page 670](#) for details.

## Search Function

The search tag is used to display a text box, with a search button on the template. When seen on the template on the Web page, the users will be allowed to enter text and click on the search button to search through all the content on the Web site.

Shown here is the format for the ecmSearch tag.

```
<CFMODULE template="#request.ecm.AppPath#ecmSearch.cfm"
 StartingFolder=" "
 Recursive=" "
 TargetPage=" "
 TextBoxSize=" "
 MaxCharacters=" "
 ButtonImageSrc=" "
 ButtonText=" "
 FontFace=" "
 FontColor=" "
 FontSize=" "
 Horizontal=" "
>
```

This function is basically the same as the Search function in ASP. See ["Search Functions" on page 621](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### Search Tag Example

```
<tr>
 <td>
 <CFMODULE template="#request.ecm.AppPath#ecmSearch.cfm"
 StartingFolder="\marketing" Recursive="1" TargetPage="search-
 display.cfm" TextBoxSize="25" MaxCharacters="200"
 ButtonImageSrc=" " ButtonText="Search" FontFace="Verdana"
 FontColor="##808080" FontSize="2" Horizontal="0">
 </td>
 </tr>
```

## Search Display Function

The search display custom tag created the area where all the search results show after the search has been executed.

Shown here is the format of the ecmSearchDisplay tag:

```
<CFMODULE template="#request.ecm.AppPath#ecmSearchDisplay.cfm"
 MaxNumber=" "
 StyleInfo=" "
 ShowDate=" "
>
```

This function is basically the same as the Search Display function in ASP. See ["Search Display Function" on page 631](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### Search Display Example

```
<tr>
 <td>
 <CFMODULE template="#request.ecm.AppPath#ecmSearchDisplay.cfm"
 MaxNumber="0" StyleInfo=" " ShowDate="1">
 </td>
</tr>
```

## List Summary Function

The list summary function is used to display the list of content summaries on a Web page. Shown here is the format of the list summary function:

```
<CFMODULE template="#request.ecm.AppPath#ecmListSummary.cfm"
 Folder=" "
 MaxNumber=" "
 Recursive=" "
 ShowSummary=" "
 StyleInfo=" "
 OrderBy=" "
 ShowInfo=" "
>
```

This function is basically the same as the List Summary function in ASP. See ["List Summary Function" on page 635](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### List Summary Example

```
<TR>
 <TD>
 <CFMODULE template="#request.ecm.AppPath#ecmListSummary.cfm"
 Folder="\Marketing\News" MaxNumber="0" Recursive="0"
 ShowSummary="1" StyleInfo=" " OrderBy="Title,asc" ShowInfo=" " >
 </TD>
</TR>
```

## XML List Summary Function

The XML List Summary function wraps XML around the properties of a list of content, created as a list summary in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the list summary information. The `ecmListSummaryXML` function allows you to easily define how the list summary data appears on the Web page.

```
<TR>
 <TD>
 <CFMODULE template="#request.ecm.AppPath#ecmListSummaryXML.cfm"
 Folder=" " Recursive=" " OrderBy=" "
 MaxNumber=" " ObjType=" " SummaryType=" " >
 </TD>
</TR>
```

This function is basically the same as the List Summary XML function in ASP. See ["XML List Summary Function" on page 642](#) for details.

The only difference between the ColdFusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### XML List Summary Example

```
<TR>
 <TD>
 <CFMODULE template="#request.ecm.AppPath#ecmListSummaryXML.cfm"
 Folder="\books\" Recursive="0" OrderBy="Title,asc"
 MaxNumber="0" ObjType="folder" SummaryType="alltypes">
 </TD>
</TR>
```

## RSS List Summary Function

The RSS summary function is used to create an XML list of content summaries for RSS Syndication. Shown here is the format of the RSS list summary function:

```
<CFMODULE template="#request.ecm.AppPath#ecmListSummary.cfm"
 Folder=" "
 MaxNumber=" "
 Recursive=" "
 OrderBy=" "
>
```

This function is basically the same as the RSS Summary function in ASP. See ["RSS Summary Function" on page 652](#) for details.

The only difference between the ColdFusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### List Summary Example

```
<TR>
 <TD>
 <CFMODULE template="#request.ecm.AppPath#ecmListSummary.cfm"
 Folder="\books" MaxNumber="0" Recursive="0"
 OrderBy="Title,asc" >
```

```
</TD>
</TR>
```

## Archive List Summary Function

The Archive list summary function displays on a Web page a list of content summaries for archived content blocks. For example, you might have a current news page and an archived news page. Use the List Summary Function to display current news, and this function to display older news stories.

On the Content Block **schedule** screen, the user must set an **End Date** then choose the Archive and Display option to have the content blocks appear via this function.

This function has the same parameters as the List Summary Function. See ["List Summary Function" on page 700](#) for details.

## Single Summary Function

The single summary tag is used to display the summary of one content block in the Ektron CMS400.NET Web site, instead of a list of all the summaries in a content folder. Seen below is the single summary tag:

```
<cfmodule template="#request.ecm.AppPath#ecmSingleSummary.cfm"
 ContentID=" "
 ShowSummary="1"
 StyleInfo=" "
 ShowInfo=" "
>
```

This function is basically the same as the Single Summary function in ASP. See ["Single Summary Function" on page 648](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

### Single Summary Example

```
<TR>
 <TD>
 <cfmodule template="#request.ecm.AppPath#ecmSingleSummary.cfm"
 ContentID="14" ShowSummary="1" StyleInfo=" "
 ShowInfo="DateModified">
 </TD>
 </TR>
```

## Collection Function

The collection function is used to display the list of links that were created as a collection in the Ektron CMS400.NET Workarea. The ecmCollection function is highly customizable, allowing you to easily define how the collection data will be displayed on the Web page.

Shown here is the format of the ecmCollection function.

```
<CFMODULE template="#request.ecm.AppPath#ecmCollection.cfm"
```

```

id="2"
displayFunction="ecmTeaser"
>

```

This function is basically the same as the Collection function in ASP. See ["Collection Function" on page 657](#) for details.

The only difference between the ColdFusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

## Example Using the ecmNavigation Display Function

The following is an example of a collection function being used as a navigation menu in the Ektron CMS400.NET sample Web site.

```

<CFMODULE id="1" displayFunction="ecmNavigation"
template="#request.ecm.AppPath#ecmCollection.cfm">

```

As stated earlier, the format of the collection on the Web page depends on the displayFunction that is used. Below is the source code for the display function "ecmNavigation".

```

<cfparam name="Attributes.nId" default="">
<cfparam name="Attributes.href" default="">
<cfparam name="Attributes.target" default="">
<cfparam name="Attributes.startStyle" default="">
<cfparam name="Attributes.endStyle" default="">
<cfset LocalError = "">
<cfobject class="CMS400.NETContentBrv2.CMS400.NETContent" Action="Create" name="cmsObj">
<cfset cNavs =
cmsObj.GetEcmCollectionItems(request.ecm.AppConfStr,Attributes.nId,0,request.ecm.cmsPreview,r
equest.ecm.user_id,request.ecm.site_id,LocalError)>
<cfif cNavs.Count()>
 <cfset nString = "">
 <cfloop collection=#cNavs# Item=nav>
 <cfset nString = nString & "
 <cfif len(Attributes.href)>
 <cfif FindNoCase("?", Attributes.hrefs)>
 <cfset nString = nString & Attributes.href & "&id=" &
 nav.Item("ContentID")>
 <cfelse>
 <cfset nString = nString & Attributes.href & "?id=" &
 nav.Item("ContentID")>
 </cfif>
 </cfif>
 <cfset nString = nString & nav.Item("ContentLinks")>
 </cfif>
 <cfset nString = nString & "" " ">
 <cfif len(Attributes.target)>
 <cfset nString = nString & "target="" " & Attributes.target & "" " ">
 </cfif>
 <cfset nString = nString & ">">
 <cfif len(Attributes.startStyle)>
 <cfset nString = nString & Attributes.startStyle>
 </cfif>
 <cfset nString = nString & nav.Item("ContentTitle")>
 <cfif len(Attributes.endStyle)>
 <cfset nString = nString & Attributes.endStyle>

```

```

 </cfif>
 <cfset nString = nString & "">
 </CFLOOP>
 <cfmodule str="#nString#" nId="#Attributes.nId#"
template="#request.ecm.AppPath#ecmshowCollectionsContent.cfm">
</cfif>
<cfset nNavs = "Nothing">
<cfset cmsObj = "Nothing">

```

As you can see, the `ecmNavigation` function is a simple ASP function that creates a table of the links that belong to the collection specified. The function also displays the title of the content blocks by using the "info("ContentTitle)") building block.

## Example Using the `ecmTeaser` Display Function

The following is an example of a collection function being used as a navigation menu in the Ektron CMS400.NET sample Web site.

```

<CFMODULE id="2" displayFunction="ecmTeaser"
template="#request.ecm.AppPath#ecmCollection.cfm">

```

In this example, the collection with an ID=2 is displayed, and the function "ecmTeaser" defines how to display the collection data on a Web page. Here is how the collection would appear on a Web page.



As stated earlier, the format of the collection on the Web page depends on the `displayFunction` that is used. Below is the source code for the display function "ecmTeaser".

```

<cfparam name="Attributes.nID" default="">
<cfparam name="Attributes.href" default="">
<cfparam name="Attributes.ShowSummary" default="">
<cfparam name="Attributes.StyleInfo" default="">
<cfparam name="Attributes.target" default="">
<cfset ErrorString = "">
<cfobject class="CMS400.NETContentBrv2.CMS400.NETContent" Action="Create" name="tObj">

```

```

<cfset cTeasers = tObj.GetEcmCollectionItems(request.ecm.AppConfStr, Attributes.nId, 0,
request.ecm.cmsPreview, request.ecm.user_id, request.ecm.site_id, ErrorString)>
<cfset tString = "<table border=""0"" cellspacing=""0"" cellpadding=""0"" width=""100%"">">
<cfif cTeasers.Count()>
 <cfloop collection=#cTeasers# Item=cT>
 <cfset tString = tString & "<tr><td>">
 <cfset tString = tString & "">
 <cfif len(Attributes.href)>
 <cfif FindNoCase("?", Attributes.hrefs)>
 <cfset tString = tString & Attributes.href & "&id=" &
 cT.Item("ContentID")>
 <cfelse>
 <cfset tStringi = tString & Attributes.href & "?id=" &
 cT.Item("ContentID")>
 </cfif>
 <cfelse>
 <cfset tString = tString & cT.Item("ContentLinks")>
 </cfif>
 <cfset tString = tString & "" " ">
 <cfif len(Attributes.target)>
 <cfset tString = tString & "target="" & Attributes.target & "" " ">
 </cfif>
 <cfset tString = tString & ">">
 <cfset tString = tString & cT.Item("ContentTitle") & "" & " (" &
 DateFormat(cT.Item("DateModified"), "mm/dd/yyyy") & " " &
 TimeFormat(cT.Item("DateModified"), "h:mm:ss tt") & ")</td></tr><tr><td>" &
 cT.Item("ContentTeaser")>
 <cfset tString = tString & "</td></tr><tr><td> </td></tr>">
 </cfloop>
 <cfset tString = tString & "</table>">
 <cfmodule str="#tString#" nId="#Attributes.nId#"
template="#request.ecm.AppPath#ecmshowCollectionsContent.cfm">
</cfif>

```

As you can see, the `ecmTeaser` function is a simple ASP function that creates a table of the links that belong to the collection specified.

The `ecmTeaser` function also displays the:

- Content Title
- Date Modified
- Content Teaser

## Collection Variables

See ["Collection Attributes" on page 660](#)

## DisplayFunction

The display function defines how collection data appears on a Web page. Using ColdFusion scripting, you can create your own display functions, or use the three included functions.

## XML Collection Function

The XML collection function wraps XML around the parameters of a list of content that's created as a collection in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the collection information. The `ecmCollectionXML` function allows you to easily define how the collection data appears on the Web page.

Shown here is the format of the `ecmCollection` function.

```
<CFMODULE template="#request.ecm.AppPath#ecmCollectionXML.cfm"
 id=" "
 getHtml=" "
>
```

This function is basically the same as the XML Collection function in ASP. See ["XML Collection Function" on page 663](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

```
<CFMODULE template="#request.ecm.AppPath#ecmCollectionXML.cfm" id="3" getHtml="0">
```

## RSS Collection Function

The RSS collection function is used to create an XML list of links that were created as a collection in the Ektron CMS400.NET Workarea and display them in RSS Syndication.

Shown here is the format of the `ecmRssCollection` function.

```
<CFMODULE template="#request.ecm.AppPath#ecmRssCollection.cfm"
 id="2"
>
```

This function is basically the same as the RSS Collection function in ASP. See ["RSS Collection Function" on page 661](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it.

## Event Calendar Function

The `ecmEvtCalendar` function is used to display an event calendar that has been created in the Ektron CMS400.NET Workarea.

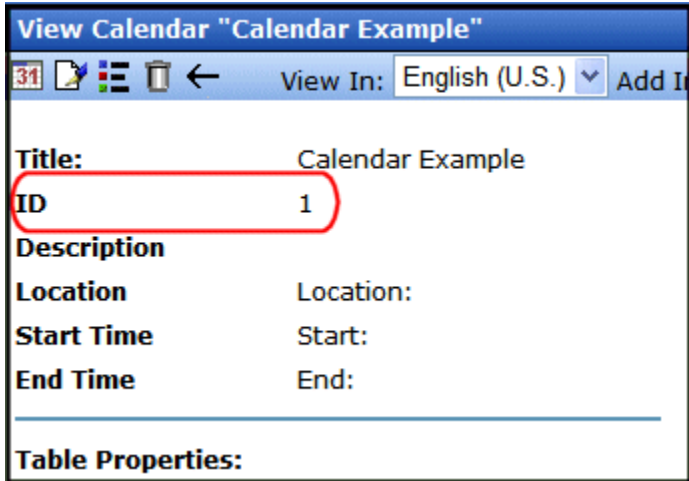
Displayed here is the format for the event calendar custom function:


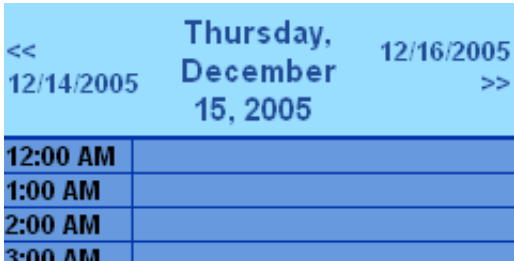
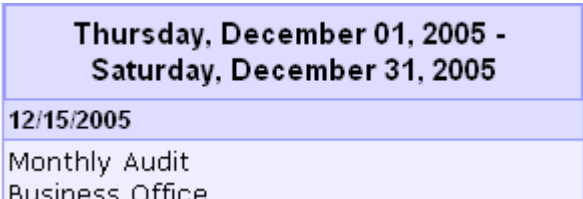
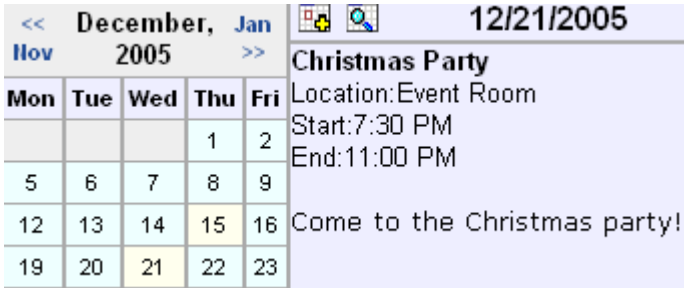
```
<CFMODULE template="#request.ecm.AppPath#ecmEvtCalendar.cfm" Calendar_ID=" " DisplayType=" ">
```

This function is basically the same as the Event Calendar function in ASP. See ["Event Calendar Function" on page 671](#) for details.



The only differences between the Coldfusion function and the ASP one are the use of the DisplayType parameter and the syntax for implementing it. The ecmEvtCalendar function parameters are defined below.

Attribute	Description
id	<p>The “<b>id</b>” argument represents the ID number of the event calendar in the Workarea.</p>  <p>The screenshot shows a web application window titled "View Calendar 'Calendar Example'". It has a toolbar with icons for calendar, print, settings, and navigation. A dropdown menu shows "View In: English (U.S.)". Below the toolbar, there are labels for "Title:", "ID", "Description", "Location", "Start Time", and "End Time". The "ID" field is highlighted with a red circle and contains the value "1". The "Title" field contains "Calendar Example". The "Description", "Location", "Start Time", and "End Time" fields are empty. At the bottom, there is a section labeled "Table Properties:".</p>

Attribute	Description
DisplayType	<p>Specify the Calendar display type.</p> <ul style="list-style-type: none"> <li> <b>Month</b> - A 30 day calendar month view.            </li> <li> <b>Day</b> - Single day view.            </li> <li> <b>Inline</b> - Display events that occur from the start date of the calendar through the end date.            </li> <li> <b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.            </li> </ul>

Here is a sample of the Event Calendar syntax.

## Event Calendar Sample

```
<tr>
 <td>
 <CFModule template="#request.ecm.AppPath#ecmEvtCalendar.cfm" calendar_id="24"
 DisplayType="monthupwiththeevent">
 </td>
</tr>
```

## Form Function

The ecmFormBlock function displays an HTML form content block. For more information about HTML forms, see the **Ektron CMS400.NET** Administrator manual chapter "Working with HTML Forms."

### IMPORTANT!

If you create a template for an existing form content block, you must manually change its quicklink to point to the new template. This change does not occur automatically. This procedure is described in the Ektron CMS400.NET Administrator manual chapter "Managing Library Assets", section "Updating Default Template for Multiple Quicklinks."

Here is the format for the form custom function

```
<CFMODULE template="#request.ecm.AppPath#ecmFormBlock.cfm" ID=" " >
```

This function is basically the same as the Form function in ASP. See ["Form Function" on page 678](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion example is below.

Here is an example of a form content block in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <CFModule template="#request.ecm.AppPath#ecmFormBlock" form_id="1">
 </td>
</tr>
```

## MultiPurpose Function

The ecmMultiPurpose function displays either a content block or an content block associated with a form. As a result, you can use one template to display both types of content block.

Here is the format for the Multipurpose function

```
<CFMODULE template="#request.ecm.AppPath#ecmMultiPurpose.cfm" DefaultID=" " >
```

To pass in a standard content block, pass the id value in the url parameter, as illustrated below.

```
http://localhost/cms400Min/index.asp?id=2
```

To pass in a Form content block, pass the ekfrm value in the url parameter, as illustrated below.

`http://localhost/cms400Min/index.asp?ekfrm=2`

**NOTE** For more information about passing the id value as a URL parameter, see ["Dynamic Content Block" on page 697](#).

If the id value or ekfrm value does not exist, a blank template appears.

If the url parameter is not passed in or recognized, the function calls `ecmContentBlock` and passes the `DefaultID` value as the id argument. If the `DefaultID` value is zero (0), `ecmContentBlock` is not called.

## Metadata Function

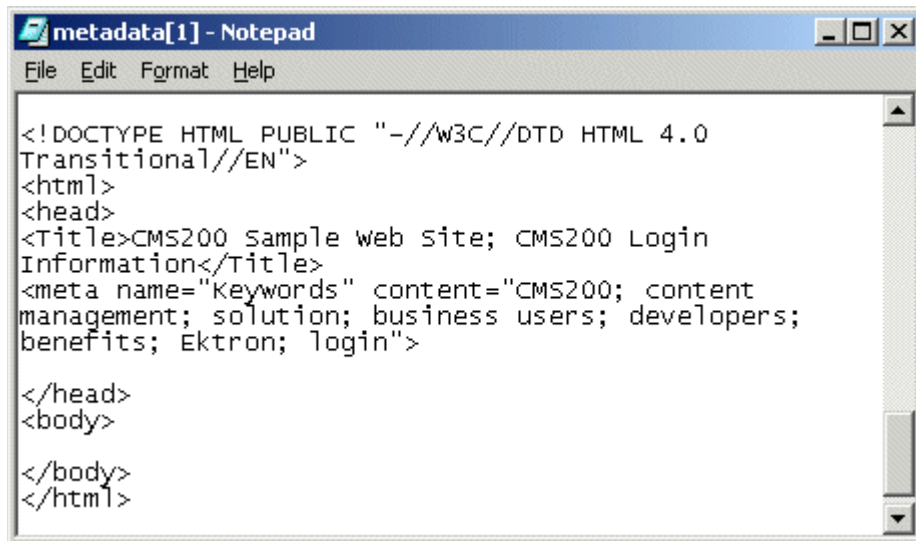
The metadata tag is used to display the metadata for one or more content blocks in the Ektron CMS400.NET Web site. Inserting this custom tag will allow you to generate the list of metadata that is edited by the authors, and insert it in the source of the Web page. Shown below is an example of what the metadata custom tag looks like.

```
<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm" ItemList="" >
```

This function is basically the same as the Metadata function in ASP. See ["Metadata Function" on page 681](#) for details.

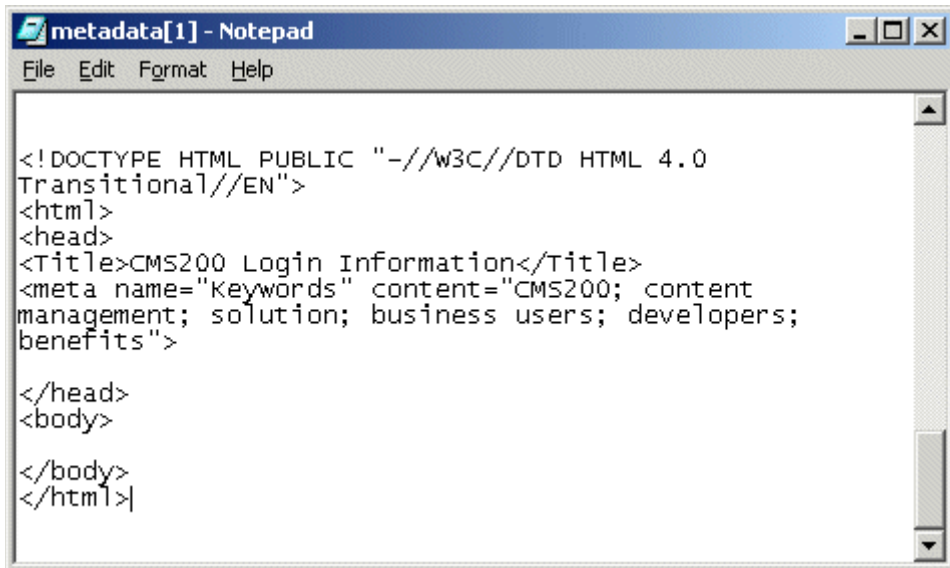
The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The ColdFusion examples are below.

```
<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm" ItemList="[1][2]">
```



```
<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm"
```

```
ItemList="[1;title][2;keywords]>
```



```

File Edit Format Help

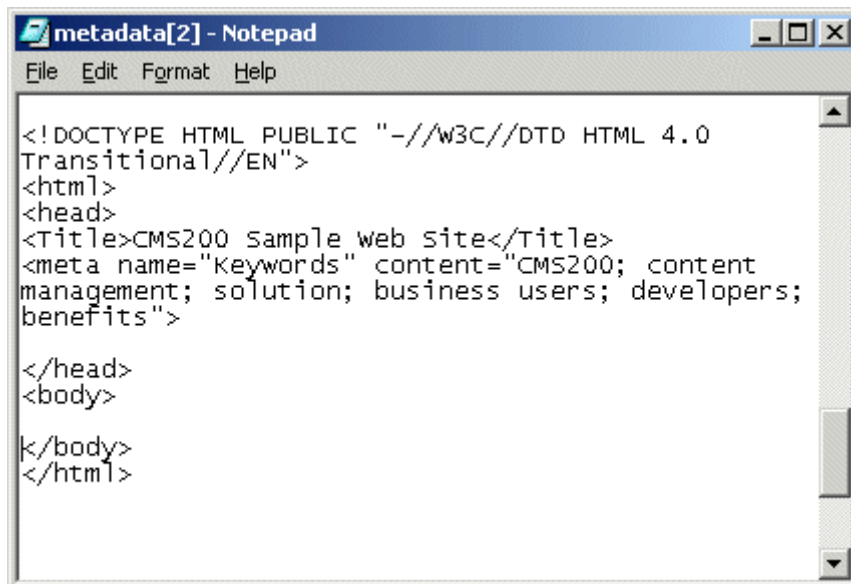
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 Login Information</Title>
<meta name="Keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

</body>
</html>

```

```
<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm"
ItemList="[1][2;title,keywords]>
```



```

File Edit Format Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 sample web site</Title>
<meta name="Keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

</body>
</html>

```

## DHTML Menus

**Ektron CMS400.NET's** Menu feature lets users create and maintain a dropdown menu system for your Web site. The Menu feature lets you display menu options that link to content blocks, library assets, external hyperlinks and submenus.

To learn more about menus, please refer to the “Menu Feature” chapter of the **Ektron CMS400.NET** User Manual.

## Additional Documentation in Ektron Knowledge Base

This chapter explains how to implement standard menus that ship with **Ektron CMS400.NET**. Ektron's Web site also provides Knowledge Base articles that explain how to extend the standard menu features.

## Displaying a Menu on a Web Page

### *GenerateMenu and DropMenu*

**IMPORTANT!** The menu functions described below are only one example of how to implement menus. You can implement your own menu system using JavaScript.

Use the `GenerateNextGenMenu` and `DropNextGenMenu` custom functions to display a DHTML menu on a Web page. Place `GenerateNextGenMenu` once on the page to provide an infrastructure for `DropNextGenMenu`. Then, place `DropNextGenMenu` as needed to implement individual menus.

Follow these steps to display a menu on a Web template.

1. Place the following code directly after the page's `<body>` tag.

```
<CFMODULE Ids="menu id number"
template="#request.ecm.AppPath#ecmDHTML_GenerateNextGenMenu.cfm">
```

For example:

```
<CFMODULE Ids="1" template="#request.ecm.AppPath#ecmDHTML_GenerateNextGenMenu.cfm">
```

To display more than one menu, separate each id number with a comma. For example:

```
<CFMODULE Ids="1,2,3" template="#request.ecm.AppPath#ecmDHTML_GenerateNextGenMenu.cfm">
```

2. If the user clicks *text* to display the menu, place the following code within the template file where the text would appear.

```
<CFMODULE MenuId="menu id number" Title="text user clicks to display menu"
template="#request.ecm.AppPath#ecmDHTML_DropNextGenMenu.cfm">
```

For example:

```
<CFMODULE MenuId="1" Title="Products"
template="#request.ecm.AppPath#ecmDHTML_DropNextGenMenu.cfm">
```

If the user clicks *a graphic* to invoke the menu, use this syntax:

```
<CFMODULE MenuId="menu id number" Title= ''
template="#request.ecm.AppPath#ecmDHTML_DropNextGenMenu.cfm">
```

For example:

```
<CFMODULE MenuId="1" Title= ''
template="#request.ecm.AppPath#ecmDHTML_DropNextGenMenu.cfm">
```

## Customizing the Menu's Appearance

You can customize a menu's appearance, such as the font style, color, and size. So, you have the flexibility to coordinate a menu with the design your Web site.

To customize a menu in **Ektron CMS400.NET**, you can change the `ek_Menu.js` file.

### **`ek_Menu.js`**

You can change the look and feel of a menu by changing the `ek_menu.js` file. If you used the default settings when installing **Ektron CMS400.NET**, this file is in `webroot/cms400Min/Workarea/java/`. For a description of the `ek_menu.js` file, see ["ek\\_Menu.js" on page 689](#).

## XML Menu Function

The `ecmGetMenuXML` function wraps XML around the properties of a menu created in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the menu. The `ecmGetMenuXML` function allows you to easily define how the menu data appears on the Web page.

```
<CFMODULE id="" template="#request.ecm.AppPath#ecmGetMenuXML.cfm">
```

This function is basically the same as the XML Menu function in ASP. See ["XML Menu" on page 692](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it. The Coldfusion example is below.

```
<CFMODULE id="1" template="#request.ecm.AppPath#ecmGetMenuXML.cfm">
```

## Language Select Function

This function lets a visitor to your Web site select a language in which to view the site. Here is what the function looks like when published on a Web page.



Shown here is the format of the `ecmLanguageSelect` function.

```
<CFMODULE template="#request.ecm.AppPath#ecmLanguageSelect.cfm">
```

This function is basically the same as the Show Site Language function in ASP. See ["Show Site Language Function" on page 687](#) for details.

The only difference between the Coldfusion function and the ASP one is the syntax for implementing it.

# Custom PHP Functions

Ektron CMS400.NET maintains and displays content via custom Ektron PHP functions and ComObjects. For the custom PHP functions to work properly, do the following:

- Make sure the `php_soap.dll` file is in the `c:\php` folder on the server where you installed PHP.
- Add the following line to the `php.ini` file.  
`extension=php_soap.dll`  
 The `php.ini` file is located in `C:\Windows\php.ini`.
- In the `ApplicationAPI.php` file, set the `WSDLpath`. The `applicationAPI.php` file is located by default in:  
`http://localhost/siteroot/WorkArea/platforms/PHP/applicationAPI.php`

## IMPORTANT!

`ContentWS.asmx` is the web services (WSDL) file that the custom PHP functions use to talk to **Ektron CMS400.NET**. The `WSDLpath` located in the `applicationAPI.php` file must be set to location of the WSDL file. The default `WSDLpath` is `http://localhost/siteroot/WorkArea/WebServices/ContentWS.asmx`. If you have changed the location of `ContentWS.asmx` you need to update the `WSDLpath`. For example, if your site is `www.example.com`, the `WSDLpath` would be `http://www.example.com/WorkArea/WebServices/ContentWS.asmx`.

- Add the following line to the top of each template.  
`<?php include 'applicationAPI.php'; ?>`  
 This file calls the Ektron custom functions file to allow the page to function properly. It *must* appear at the top of the template file.
- If you have upgraded your Ektron CMS400.NET from an Ektron CMS300 version, add the following line between the head tags for each template.

## IMPORTANT!

This only needs to be done if you are upgrading from CMS300 to CMS400.NET

```
<meta http-equiv="Content-type" content="text/html; charset=utf-8" />
```

To customize you Ektron CMS400.NET application, you need to be familiar with these functions.

Function	Description	More Information
<b>Calendar Functions</b>		
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 727



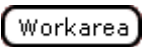




Function	Description	More Information
<b>Collection Functions</b>		
Collection	Displays links generated by collections	"Collection Function" on page 722
CollectionXML	Creates XML for a collection by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML Collection Function" on page 725
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 726
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 726
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 726
<b>Content Functions</b>		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> programmatically.	"Add Content" on page 718
Content Block	Displays and manages a content block	"Content Block Function" on page 717
Form	Displays a content block associated with a form	"Form Function" on page 729
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 730
<b>List Summary Functions</b>		
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 721
List Summary	Displays a list of content block summaries	"List Summary Function" on page 720

Function	Description	More Information
List Summary XML	Creates XML for a list summary by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML List Summary Function" on page 722
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 722
Single Summary	Displays the summary for a content block	"Single Summary Function" on page 721
<b>Login Functions</b>		
Login	Adds the login or logout button	"Login Function" on page 717
<b>Menu Functions</b>		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 732
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 732
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 734
<b>Metadata Functions</b>		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 730
<b>Search Functions</b>		
Search	Inserts a text box with a search button	"Search Function" on page 719
Search Display	Creates an area to display search results	"Search Display Function" on page 720
<b>Language Functions</b>		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 734

## Login Function

This function is responsible for displaying the following buttons on the Web page.

Button	Name	Description
	Login	When user is not logged into Ektron CMS400.NET, this button appears. Click this button to let the user log in to the application.
	Logout	Once logged in, this button replaces the login button to allow the user to log out from the Ektron CMS400.NET Web site.
	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.
	Preview Off	Turns off preview mode.

This function is basically the same as the Login function in ASP. See "[Login Function](#)" on page 607 for details. The only difference is the syntax used to implement the function. This is listed below.

### Login Function Example

```
<tr>
 <td>
 <?php ecmlogin(); ?>
 </td>
</tr>
```

## Content Block Function

The ecmContentBlock and ecmContentBlockEX functions are used to display a content block on an Ektron CMS400.NET Web page.

Shown here is the format for the ecmContentBlock function:

```
<?php ecmContentBlock (ID) ?>
```

Here is the format for the ecmContentBlockEX function:

```
<?php ecmContentBlockEx (ID, "XSLT", Override) ?>
```

**NOTE** As an alternative, you can use the [multipurpose function](#), which can display either a content block or a content block associated with a form. For more information, see ["MultiPurpose Function" on page 730](#).

This function is basically the same as the Content Block function in ASP. See ["Content Block Function" on page 609](#) for details.

The only differences between the PHP function and the ASP one are the examples of implementing it. The PHP examples are below.

## Static Content Block

```
<tr>
 <td>
 <?php ecmContentBlock(1) ?>
 </td>
</tr>
```

## Dynamic Content Block

```
<tr>
 <td>
 <?php ecmContentBlock($id) ?>
 </td>
</tr>
```

## XML Content Block

```
<tr>
 <td>
 <?php ecmContentBlockEx (13, "", 1) ?>
 </td>
</tr>
```

## Add Content

The add content function allows you to programmatically add a new content block to Ektron CMS400.NET without the need of logging in.

Here is the format for the add content function:

```
<? AddContent ("Username"
 "Password",
 "Domain",
 'content_title',
 "content_comment",
 "ContentHtml",
 "SummaryHTML",
 "ContentLanguage",
 "FolderID",
 "GoLive",
 "EndDate",
 "MetaInfo"); ?>
```

This function is basically the same as the Add Content function in ASP. See ["Add Content" on page 617](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

## Add Content Example

The following is an example of the add content function in a template:.

```
<tr>
 <td>
 <? AddContent ("admin", "admin", "", "Content Title", "Comment",
 "This is the body of the content.", "This is the <i>summary</i>
 for the content.", "1033", "0", "", "", "", ""); ?>
 </td>
</tr>
```

## Search Function

The search function is used to display a text box with a search button on the template. When seen on the template on the Web page, a user can enter text and click the search button to search through all the content on the Web site.

Ektron has created two search functions:

- ecmSearch
- ecmSearchNoFormTag

Both functions have the same parameters and perform the same search. The only difference is ecmSearch automatically inserts form tags, ecmSearchNoFormTag does not.

The following is an example of form tags.

```
<form name="ecmsearch" method="post" action='search-
display.asp'>
```

```
</form>
```

When using ecmSearch, form tags are added automatically. When using ecmSearchNoFormTag you need to add the tags manually

Here is the format for the search function. You can interchange ecmSearch and ecmSearchNoFormTag when using this function.

```
<?php ecmSearch("StartingFolder",
 Recursive,
 "targetpage",
 textbox,
 MaxCharacters,
 "ButtonImgSrc",
 "ButtonText",
 "FontFace",
 "FontColor",
 "FontSize",
```

```
Horizontal,
"Spare");
?>
```

This function is basically the same as the Search function in ASP. See ["Search Functions" on page 621](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

## Search Function Example

```
<TR>
 <TD>
 <?php ecmSearch("\Marketing",1,"search-
display.php",25,200,"","Search","Verdana","#808080","2",0,""); ?>
 </TD>
</TR>
```

## Search Display Function

The search display custom function created the area where all the search results show after the search has been executed.

Here is the format of the ecmSearchDisplay function:

```
<?php ecmSearchDisplay(MaxNumbers,"StyleInfo", ShowDate); ?>
```

This function is basically the same as the Search Display function in ASP. See ["Search Display Function" on page 631](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

## Search Display Function Example

```
<tr>
 <td>
 <?php ecmSearchDisplay(0,"",1); ?>
 </td>
</tr>
```

## List Summary Function

The list summary function is used to display the list of content summaries on a Web page. Here is the format of the list summary function.

```
<?php ecmListSummary ("Folder",
 Recursive,
 ShowSummary,
 "StyleInfo",
 "OrderBy",
 "ShowInfo",
 MaxNumber,
 "Spare 1",
 "Spare2");
?>
```

This function is basically the same as the List Summary function in ASP. See ["List Summary Function" on page 635](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

## List Summary Example

```
<TR>
 <TD>
 <?PHP ecmListSummary ("Marketing\News",0,1,"",
 "Title,asc","DateModified",0,"",""); ?>
 </TD>
</TR>
```

## Archive List Summary Function

The Archive list summary function displays on a Web page a list of content summaries for archived content blocks. For example, you might have a current news page and an archived news page. Use the List Summary Function to display current news, and this function to display older news stories.

On the Content Block **schedule** screen, the user must set an **End Date** then choose the Archive and Display option to have the content blocks appear via this function.

This function has the same parameters as the List Summary Function. See ["List Summary Function" on page 720](#) for details.

## Single Summary Function

The single summary function is used to display the summary of one content block in the Ektron CMS400.NET Web site, instead of a list of all the summaries in a content folder. Seen below is the single summary function:

```
<?php ecmSingleSummary("ContentID",
 ShowSummary,
 "StyleInfo",
 "ShowInfo",
 "Spare1");
?>
```

This function is basically the same as the Single Summary function in ASP. See ["Single Summary Function" on page 648](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

## Single Summary Example

```
<TR>
 <TD>
 <?PHP ecmSingleSummary ("14",1,"","DateModified",""); ?>
 </TD>
</TR>
```

## XML List Summary Function

The XML List Summary function wraps XML around the properties of a list of content, created as a list summary in the Ektron CMS400.NET Workarea. Once created, you can use the XML to display the list summary information. The `ecmListSummaryXML` function allows you to easily define how the list summary data appears on the Web page.

```
<?php ecmListSummaryXML ("Folder", Recursive,
"OrderBy", MaxNumber, "ObjType", "SummaryType"); ?>
```

This function is basically the same as the List Summary XML function in ASP. See ["XML List Summary Function" on page 642](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<?php ecmListSummaryXML("\books\\",0,"Title,asc",0,"folder",""); ?>
```

## RSS List Summary Function

The RSS summary function is used to create an XML list of content summaries for RSS Syndication. Shown here is the format of the RSS list summary function:

```
<?php ecmRssSummary ("Folder", Recursive, "OrderBy", MaxNumber); ?>
```

This function is basically the same as the RSS Summary function in ASP. See ["RSS Summary Function" on page 652](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<?php ecmRssSummary("\CMS400Min\\", 0, "Title,asc", 0); ?>
```

## Collection Function

The collection function is used to display the list of links that were created as a collection in the Ektron CMS400.NET Workarea. The `ecmCollection` function is highly customizable, allowing you to easily define how the collection data will be displayed on the Web page.

Shown here is the format of the `ecmCollection` function.

```
<?PHP ecmCollection("id",
"displayFunction";
?>
```

This function is basically the same as the Collection function in ASP. See ["Collection Function" on page 657](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.



## Collection Examples

### Example Using the *ecmNavigation Display Function*

The following is an example of a collection function being used as a navigation menu in the Ektron CMS400.NET sample Web site.

```
<?php ecmCollection(1, "ecmNavigation") ?>
```

In this example, the collection with an ID=1 is displayed, and the function "ecmNavigation" is used to define how the collection data is displayed on the Web page.

Here is how the collection would appear on the Web page.



The format of the collection on the Web page depends on the displayFunction that is used.

Below is the source code for the display function "ecmNavigation".

```

Function ecmNavigation($cInfo){
 global $html, $info;
 $html = "<table border=\"0\" cellspacing=\"0\" cellpadding=\"0\"
 width=\"100%\">";
 $mycount = $cInfo->count();
 for ($iloop=0; $iloop < $mycount; ++$iloop) {
 $info = $cInfo->Item($iloop+1);
 $html = $html."<tr><td> <a href=\"";
 $html = $html.$info->Item("ContentLinks");
 $html = $html.\">\";
 $html = $html.$info->Item("ContentTitle");
 $html = $html.\"</td></tr><tr><td> </td></tr>\";
 }
 $html = $html.</table>\";
 return $html;
}

```

As you can see, the ecmNavigation function is a simple PHP function that creates a table of the links that belong to the collection specified.

The ecmNavigation function also displays the title of the content blocks by using the “info(“ContentTitle”)” building block.

### Example Using the ecmTeaser Display Function

The following is an example of a collection function being used as a navigation menu in the Ektron CMS400.NET sample Web site.

```
<?php ecmCollection (2, "MyDisplayFunction"); ?>
```

In this example, the collection with an ID=2 is displayed, and the function “ecmTeaser” is used to define how the collection data is displayed on the Web page.

Shown here is how the collection would appear on the Web page.

---

Home | Products | Support | News | **Careers** | Search

---

## Employment Opportunities



**Plastic Molder #123** (8/27/2002 2:38:34 PM)

RC International is looking for an experienced plastics molder.

**Servo Control Engineer #124** (8/27/2002 2:40:26 PM)

RC International is looked for a highly skilled servo control engineer to join our team.

As stated earlier, the format of the collection on the Web page depends on the displayFunction that is used.

Shown below is the source code for the display function “ecmTeaser”

```

Function ecmTeaser($cInfo){
 global $html, $info;
 $html = "<table border=\"0\" cellspacing=\"0\" cellpadding=\"0\" width=\"100%\">";
 $mycount = $cInfo->count();
 for ($i=0; $i < $mycount; ++$i) {
 $info = $cInfo->Item($i+1);
 $html = $html."<tr><td><a href=\"";
 $html = $html.$info->Item("ContentLinks");
 $html = $html."\" >";
 $html = $html.$info->Item("ContentTitle");
 $html = $html." ";
 $html = $html.date("m/d/Y g:i:s A", $info->Item("DateModified"));
 $html = $html."</td></tr><tr><td>";
 $html = $html.$info->Item("ContentTeaser");
 $html = $html."</td></tr><tr><td> </td></tr>";
 }
 $html = $html."</table>";
 return $html;
}

```

As you can see, the ecmTeaser function is a simple ASP function that creates a table of the links that belong to the collection specified.

The ecmTeaser function also displays the:

- Content Title
- Date Modified
- Content Teaser

## Collection Variables

See ["Collection Attributes" on page 660](#)

## DisplayFunction

As explained earlier, the display function is a function that you create that will define how the collection data will be displayed on the Web page.

By using simple PHP scripting, you can create your own display functions, or use the three included functions.

## XML Collection Function

The XML collection function wraps XML around the parameters of a list of content that's created as a collection in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the collection information. The ecmCollectionXML function allows you to easily define how the collection data appears on the Web page.

Shown here is the format of the ecmCollection function.

```
<?php ecmCollectionXML (id, GetHtml); ?>
```

This function is basically the same as the XML Collection function in ASP. See ["XML Collection Function" on page 663](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<?php ecmCollectionXML (3,0); ?>
```

## RSS Collection Function

The RSS collection function is used to create an XML list of links that were created as a collection in the Ektron CMS400.NET Workarea and display them in RSS Syndication.

Shown here is the PHP format of the ecmRssCollection function.

**<?php ecmRssCollection (id); ?>**

This function is basically the same as the RSS Collection function in ASP. See ["RSS Collection Function" on page 661](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it.

## Random Content Function

The random content function is used to randomly display the content of a content block that belongs in a specified collection.

Displayed here is the format for the random content function

**<?php ShowRandomContent (id); ?>**

This function is basically the same as the Random Content function in ASP. See ["Random Content" on page 669](#) for details.

## Random Summary Function

Similar to the random content function, the random summary function is used to randomly display the summary of a content block that belongs in a specified collection.

Displayed here is the format for the random content function

**<?php ShowRandomSummary (id, XsltID); ?>**

This function is basically the same as the Random Summary function in ASP. See ["Random Summary" on page 670](#) for details.

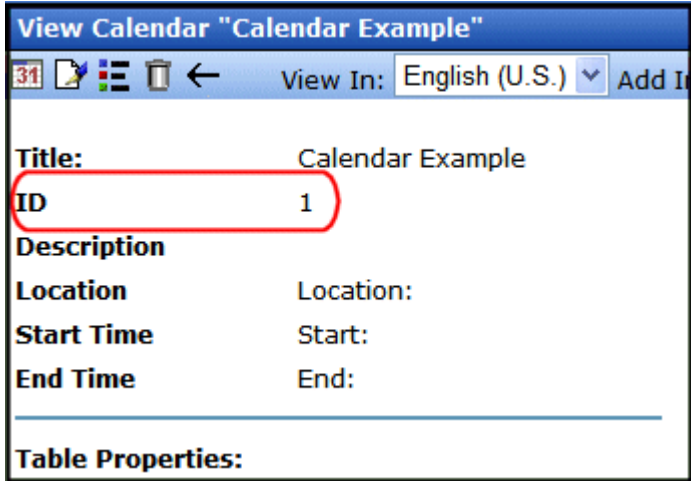
## Event Calendar Function

The ecmEvtCalendar function is used to display an event calendar that has been created in the Ektron CMS400.NET Workarea.

```
<?php ecmEvtCalendar(id, "Displaytype"); ?>
```

This function is basically the same as the Event Calendar function in ASP. See ["Event Calendar Function" on page 671](#) for details.

The only differences between the PHP function and the ASP one are the use of the DisplayType parameter and the syntax for implementing it. The ecmEvtCalendar function parameters are defined below.

Attribute	Description
id	<p>The "id" argument represents the ID number of the event calendar in the Workarea.</p> 

Attribute	Description																													
DisplayType	<p>Specify the Calendar display type.</p> <ul style="list-style-type: none"><li><b>Month</b> - A 30 day calendar month view.</li></ul> <div><div>&lt;&lt; November</div><div>December 2005</div><div>January &gt;&gt;</div><table><thead><tr><th>Monday</th><th>Tuesday</th><th>Wednesday</th><th>Thursday</th><th>Friday</th></tr></thead><tbody><tr><td></td><td></td><td></td><td>1</td><td>2</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr></tbody></table></div> <ul style="list-style-type: none"><li><b>Day</b> - Single day view.</li></ul> <div><div>&lt;&lt; 12/14/2005</div><div>Thursday, December 15, 2005</div><div>12/16/2005 &gt;&gt;</div><table><tbody><tr><td>12:00 AM</td><td></td></tr><tr><td>1:00 AM</td><td></td></tr></tbody></table></div> <ul style="list-style-type: none"><li><b>Inline</b> - Display events that occur from the start date of the calendar through the end date.</li></ul> <div><div>Thursday, December 01, 2005 - Saturday, December 31, 2005</div><div>12/15/2005</div><div>Monthly Audit</div></div> <ul style="list-style-type: none"><li><b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.</li></ul> <div><div>&lt;&lt; December, Jan</div><div>2005</div><div>&gt;&gt;</div><div>Mon Tue Wed Thu Fri</div><div>1 2</div><div>5 6 7 8 9</div><div>12 13 14 15 16</div><div>19 20 21 22 23</div><div>26 27 28 29 30</div><div>2004 .. 2005 .. 2006</div><div><div>12/21/2005</div><div>Christmas Party</div><div>Location: Event Room</div><div>Start: 7:30 PM</div><div>End: 11:00 PM</div><div>Come to the Christmas party!</div></div></div>	Monday	Tuesday	Wednesday	Thursday	Friday				1	2											5	6	7	8	9	12:00 AM		1:00 AM	
Monday	Tuesday	Wednesday	Thursday	Friday																										
			1	2																										
5	6	7	8	9																										
12:00 AM																														
1:00 AM																														

## Event Calendar Sample

Here is an example of the event Calendar in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <?php ecmEvtCalendar(24, "MonthUpWithEvent"); ?>
 </td>
</tr>
```

## Form Function

The `ecmFormBlock` and `ecmFormBlockNoFormTag` functions display an HTML form content block. Both functions have the same parameters and display the form the same way. The only difference is `ecmFormBlock` automatically inserts form tags, `ecmFormBlockNoFormTag` does not.

The following is an example of form tags.

```
<form name="myform" action="#" onsubmit="return
EkFmValidate(this);" method="post">

</form>
```

When using `ecmFormBlock`, form tags are added automatically. When using `ecmFormBlockNoFormTag`, form tags need to be added manually.

For more information about HTML forms, see the **Ektron CMS400.NET Administrator manual** chapter "Working with HTML Forms."

### IMPORTANT!

If you create a template for an existing form content block, you must manually change its quicklink to point to the new template. This change does not occur automatically. This procedure is described in the Ektron CMS400.NET Administrator manual chapter "Managing Library Assets", section "Updating Default Template for Multiple Quicklinks."

Here is the format for the form custom function

```
<?php ecmFormBlock(id); ?>
<?php ecmFormBlockNoForTag(id); ?>
```

This function is basically the same as the Form function in ASP. See ["Form Function" on page 678](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<tr>
 <td>
 <?php ecmFormBlock(1); ?>
 </td>
</tr>
```

---

**NOTE** As an alternative, you can use the [multipurpose function](#), which can display either a content block or a content block associated with a form. For more information, see ["MultiPurpose Function" on page 730](#).

---

## MultiPurpose Function

The `ecmMultiPurpose` function displays either a content block or a content block associated with a form. As a result, you can use one template to display both types of content block.

Here is the format for the Multipurpose function

**<?php ecmMultiPurpose(ID); ?>**

To pass in a standard content block, pass the `id` value in the `url` parameter, as illustrated below.

```
http://localhost/CMS400Min/index.asp?id=2
```

To pass in a Form content block, pass the `ekfrm` value in the `url` parameter, as illustrated below.

```
http://localhost/CMS400Min/index.asp?ekfrm=2
```

---

**NOTE** For more information about passing the `id` value as a URL parameter, see ["Dynamic Content Block" on page 718](#).

---

If the `id` value or `ekfrm` value does not exist, a blank template appears.

If the `url` parameter is not passed in or recognized, the function calls `ecmContentBlock` and passes the `DefaultID` value as the `id` argument. If the `DefaultID` value is zero (0), `ecmContentBlock` is not called.

## Metadata Function

The metadata function is used to display the metadata for one or more content blocks in the Ektron CMS400.NET Web site. Inserting this custom function allows you to generate the list of metadata that is edited by the authors, and insert it in the source of the Web page. Shown below is an example of what the metadata custom function looks like.:

```
<?php ecmMetaData ("ItemList","Spare1"); ?>
```

This function is basically the same as the Metadata function in ASP. See ["Metadata Function" on page 681](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP examples are below.

To include both content blocks' metadata in the web page, an `ecmMetadata` call would look like this.

```
<html>
<head>
 <?php ecmMetadata ("[1][2;Title,Content-Type]", ""); ?>
 (rest of head)
```



```

 </head>
 <body>
 (rest of web page document)
 </body>
</html>

```

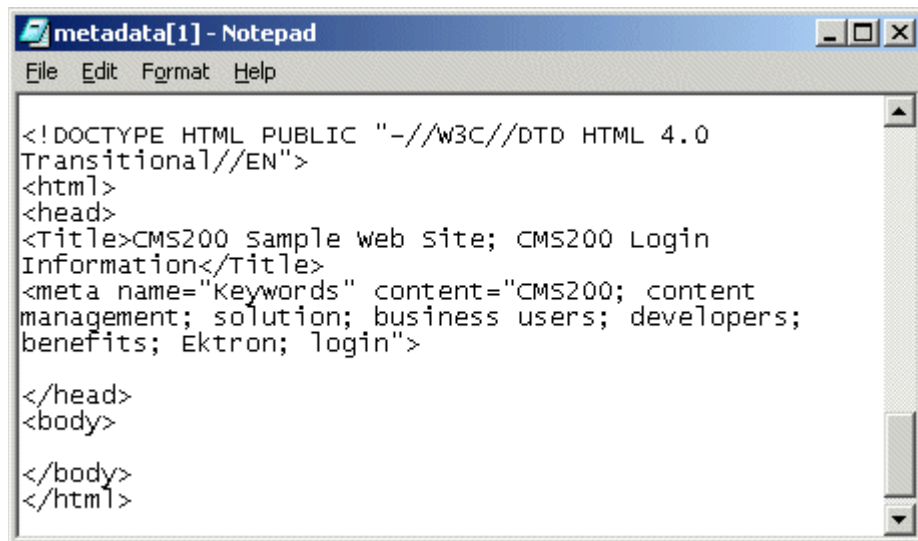
In the above example the ecmMetadata call is contained in the <head> of the HTML document (since all HTML metadata is required to exist there). Once the Web page is viewed the Web page source will contain the following information:

```

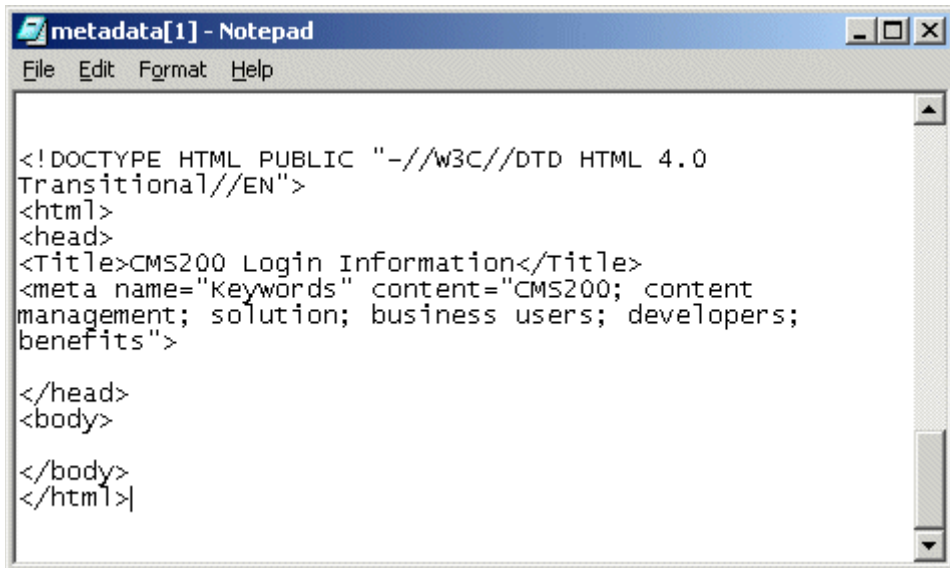
<html>
 <head>
 <Title>Welcome to Our Home Page</Title>
 <meta name="Keywords" content= Home; page; company; Welcome; Announce;
 product; widget">
 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
 (rest of head)
 </head>
 <body>
 (rest of web page document)
 </body>
</html>

```

```
<?PHP ecmMetaData ("[1][2]","") ?>
```



```
<?PHP ecmMetaData ("[1:title][2:keywords]","") ?>
```



```

File Edit Format Help

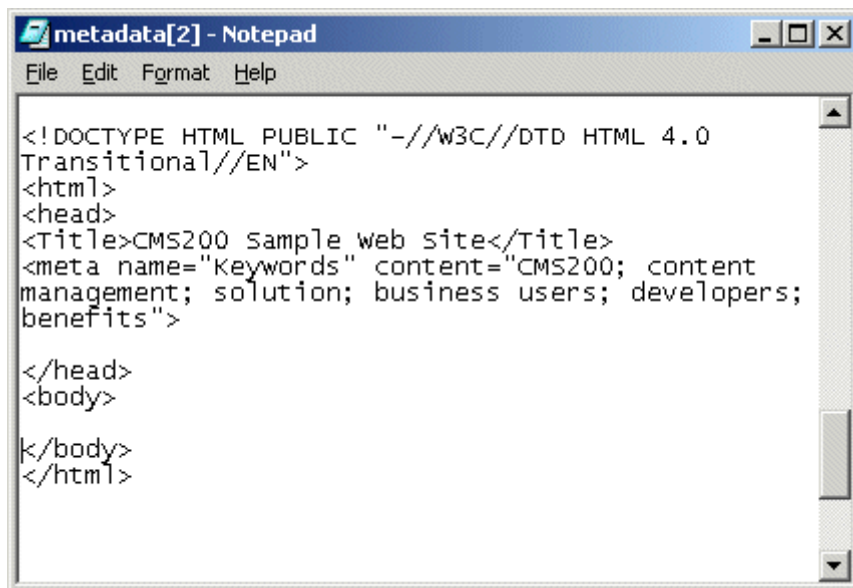
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 Login Information</Title>
<meta name="Keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

</body>
</html>

```

```
<?PHP ecmMetaData ("[1][2:title,keywords]","") ?>
```



```

File Edit Format Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 sample web site</Title>
<meta name="Keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

</body>
</html>

```

## DHTML Menus

**Ektron CMS400.NET**'s Menu feature lets users create and maintain a dropdown menu system for your Web site. The Menu feature lets you display menu options that link to content blocks, library assets, external hyperlinks and submenus.

To learn more about menus, please refer to the “Menu Feature” chapter of the **Ektron CMS400.NET** User Manual.

## Additional Documentation in Ektron Knowledge Base

This chapter explains how to implement standard menus that ship with **Ektron CMS400.NET**. Ektron's Web site also provides Knowledge Base articles that explain how to extend the standard menu features.

## Displaying a Menu on a Web Page

### *GenerateMenu and DropMenu*

**IMPORTANT!** The menu functions described below are only one example of how to implement menus. You can implement your own menu system using JavaScript.

Use the `GenerateNextGenMenu` and `DropNextGenMenu` custom functions to display a DHTML menu on a Web page. Place `GenerateNextGenMenu` once on the page to provide an infrastructure for `DropNextGenMenu`. Then, place `DropNextGenMenu` as needed to implement individual menus.

Follow these steps to display a menu on a Web template.

1. Place the following code directly after the page's `<body>` tag.

```
<?php ecmDHTML_GenerateNextGenMenu("menu id number"); ?>
```

For example:

```
<?php ecmDHTML_GenerateNextGenMenu("1"); ?>
```

To display more than one menu, separate each id number with a comma. For example:

```
<?php ecmDHTML_GenerateNextGenMenu("1,2,3"); ?>
```

2. If the user clicks *text* to display the menu, place the following code within the template file where the text would appear.

```
<?php ecmDHTML_DropNextGenMenu menu ID number,"text user clicks to display menu"; ?>
```

For example:

```
<?php ecmDHTML_DropNextGenMenu 1,"MyMenu1"; ?>
```

If the user clicks *a graphic* to invoke the menu, use this syntax:

```
<?php ecmDHTML_DropNextGenMenu menu ID number, ''; ?>
```

For example:

```
<?php ecmDHTML_DropNextGenMenu 1,''; ?>
```

## Customizing the Menu's Appearance

You can customize a menu's appearance, such as the font style, color, and size. So, you have the flexibility to coordinate a menu with the design your Web site.

To customize a menu in **Ektron CMS400.NET**, you can change the `ek_Menu.js` file.

**ek\_Menu.js**

You can change the look and feel of a menu by changing the ek\_menu.js file. If you used the default settings when installing **Ektron CMS400.NET**, this file is in `webroot/siteroot/Workarea/java/`. For a description of the ek\_menu.js file, see ["ek\\_Menu.js" on page 689](#).

## XML Menu Function

The ecmGetMenuXML function wraps XML around the properties of a menu created in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the menu. The ecmGetMenuXML function allows you to easily define how the menu data appears on the Web page.

```
<?php ecmGetMenuXML($Id); ?>
```

This function is basically the same as the XML Menu function in ASP. See ["XML Menu" on page 692](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<?php ecmGetMenuXML(1); ?>
```

## Language Select Function

This function lets a visitor to your Web site select a language in which to view the site. Here is what the function looks like when published on a Web page.



Shown here is the format of the ecmLanguageSelect function.

```
<?php ecmLanguageSelect(); ?>
```

This function is basically the same as the Show Site Language function in ASP. See ["Show Site Language Function" on page 687](#) for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it.

# Custom API

As you have seen, the formatting for the custom functions used in Ektron CMS400.NET for PHP are defined in the file `ApplicationAPI.php`. As a developer, you may make changes to this file to further enhance how the custom functions appear on your Web site.

However, Ektron recommends creating a copy of the file, and renaming it to an arbitrary name, for example, `MyCustomAPI.php`, and including that file when creating Ektron CMS400.NET templates. This ensures that your custom API will not be overwritten if you upgrade to another version of Ektron CMS400.NET.

# Custom JSP Functions

Ektron CMS400.NET maintains and displays content via custom Ektron JSP functions. For the custom JSP functions to work properly, do the following:

- In the `applicationapi.jsp` file, set the `WSSoap_address`. The `applicationapi.jsp` file is located by default in:  
`http://localhost/siteroot/WorkArea/platforms/JSP/applicationapi.jsp`

## IMPORTANT!

`ContentWS.asmx` is the web services file that the custom JSP functions use to talk to **Ektron CMS400.NET**. The `WSSoap_address` located in the `applicationapi.jsp` file must be set to location of the Web services file. The default `WSSoap_address` is `http://localhost/siteroot/WorkArea/WebServices/ContentWS.asmx`. If you have changed the location of `ContentWS.asmx` you need to update the `WSSoap_address`. For example, if your site is `www.example.com`, the `WSSoap_address` would be `http://www.example.com/WorkArea/WebServices/ContentWS.asmx`.

- Add the following line to the top of each template.  

```
<%@ include file="applicationapi.jsp" %>
```

This file calls the Ektron custom functions file to allow the page to function properly. It *must* appear at the top of the template file.
- Make sure the `contentws.jar` file is included in your JSP environment
- Ektron recommends running Axis when using a JSP environment. We have tested on TomCat and JRUN with Axis v1.2.1. You can download Axis from <http://ws.apache.org/axis/java/>.

To customize you Ektron CMS400.NET application, you need to be familiar with these functions.

Function	Description	More Information
<b>Calendar Functions</b>		
Calendar	Displays a calendar with different display types. For example, a daily view calendar. For more information on the types of calendars that can be displayed, see <a href="#">"DisplayType" on page 675</a>	<a href="#">"Calendar Function" on page 738</a>
Event Calendar	Displays a month view event calendar	<a href="#">"Event Calendar Function" on page 739</a>
<b>Collection Functions</b>		

Function	Description	More Information
Collection	Displays links generated by collections	"Collection Function" on page 740
CollectionXML	Creates XML for a collection by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML Collection Function" on page 740
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 741
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 741
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 741
<b>Content Functions</b>		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> programmatically.	"Add Content" on page 742
Content Block	Displays and manages a content block	"Content Block Function" on page 742
Form	Displays a content block associated with a form	"Form Function" on page 743
<b>Language Functions</b>		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 745
<b>Login Functions</b>		
Login	Adds the login or logout button	"Login Function" on page 745
<b>Menu Functions</b>		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 746

Function	Description	More Information
GenerateMenu	Works with DropHTMLMenu to display DHTML menus.	"DHTML Menus" on page 746
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 747
<b>Metadata Functions</b>		
Metadata	Displays the metadata for the current Web page.	"Metadata Function" on page 748
<b>Search Functions</b>		
Search	Inserts a text box with a search button.	"Search Function" on page 750
Search Display	Creates an area to display search results.	"Search Display Function" on page 751
<b>Summary Functions</b>		
List Summary	Displays a list of content block summaries.	"List Summary Function" on page 752
List Summary XML	Creates XML for a list summary by wrapping tags around all of the collection items and their properties. This allows you to use an XSLT to define how the XML collection is displayed.	"XML List Summary Function" on page 752
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 752
Single Summary	Displays the summary for a content block.	"Single Summary Function" on page 753

## Calendar Function

The `ecmCalendar` function displays a calendar that has been created in Ektron CMS400.NET. Here is the format for the event calendar custom function.

```
<%= ecmCalendar (int calendarID, String "displayType") %>
```



---

**NOTE** While you can display a month view calendar with `ecmCalendar`, you can also use `ecmEvtCalendar` to display a month view calendar. For more information, see ["Event Calendar Function" on page 739](#)

---

This function is basically the same as the Calendar function in ASP. See ["Calendar Function" on page 674](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it.

## Calendar Sample

Here is an example of the Calendar function in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <%= ecmEvtCalendar(24, "MonthUpWithEvent") %>
 </td>
</tr>
```

## Event Calendar Function

The `ecmEvtCalendar` function is used to display an event calendar that has been created in the Ektron CMS400.NET Workarea.

```
<%= ecmEvtCalendar(int calendar id) %>
```

This function is basically the same as the Event Calendar function in ASP. See ["Event Calendar Function" on page 671](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it.

## Event Calendar Sample

Here is an example of the event Calendar in an Ektron CMS400.NET template.

```
<tr>
 <td>
 <%= ecmEvtCalendar(24) %>
 </td>
</tr>
```

## Collection Function

The collection function is used to display the list of links that were created as a collection in the Ektron CMS400.NET Workarea. The `ecmCollection` function is highly customizable, allowing you to easily define how the collection data will be displayed on the Web page.

Shown here is the format of the `ecmCollection` function.

```
<%= ecmCollection(int id, String "displayFunction") %>
```

This function is basically the same as the `Collection` function in ASP. See ["Collection Function" on page 657](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it.

### Collection Sample

Here is an example of the `Collection` function in an Ektron CMS400.NET template.

```
<tr>
 <td>

 <%= ecmCollection(24, "ecmTeaser") %>

 </td>
</tr>
```

## XML Collection Function

The XML collection function wraps XML around the parameters of a list of content that's created as a collection in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the collection information. The `ecmCollectionXML` function allows you to easily define how the collection data appears on the Web page. Shown here is the format of the `ecmCollectionXML` function.

```
<%= ecmCollectionXML (int id, boolean GetHtml) %>
```

This function is basically the same as the XML Collection function in ASP. See ["XML Collection Function" on page 663](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

```
<tr>
 <td>

 <%= ecmCollectionXML(24, True) %>

 </td>
</tr>
```

## Random Content Function

The random content function is used to randomly display the content of a content block that belongs in a specified collection.

Displayed here is the format for the random content function

```
<%= ecmShowRandomContent (int collectionID, int xsId) %>
```

This function is basically the same as the Random Content function in ASP. See ["Random Content" on page 669](#) for details.

## Random Summary Function

Similar to the random content function, the random summary function is used to randomly display the summary of a content block that belongs in a specified collection.

Displayed here is the format for the random content function

```
<%= ShowRandomSummary (int collectionID) %>
```

This function is basically the same as the Random Summary function in ASP. See ["Random Summary" on page 670](#) for details.

## RSS Collection Function

The RSS collection function is used to create an XML list of links that were created as a collection in the Ektron CMS400.NET Workarea and display them in RSS Syndication.

Shown here is the JSP format of the ecmRssCollection function.

```
<%= ecmRssCollection (int id) %>
```

This function is basically the same as the RSS Collection function in ASP. See ["RSS Collection Function" on page 661](#) for details.

## Add Content

The add content function allows you to programmatically add a new content block to Ektron CMS400.NET without the need of logging in.

Here is the format for the add content function:

```
<%= ecmAddContent (String UserName, String Password, String Domain, String content_title,
String content_comment, String ContentHtml, String SummaryHTML, String ContentLanguage,
String FolderID, String GoLive, String EndDate, String MetaInfo) %>
```

This function is basically the same as the Add Content function in ASP. See ["Add Content" on page 617](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

### Add Content Example

The following is an example of the add content function in a template:.

```
<tr>
 <td>
 <%= ecmAddContent ("admin", "admin", "", "Content Title",
"Comment", "This is the body of the content.", "This is the
<i>summary</i> for the content.", "1033", "0", "", "", "") %>
 </td>
</tr>
```

## Content Block Function

The ecmContentBlock and ecmContentBlockEX functions are used to display a content block on an Ektron CMS400.NET Web page.

Shown here is the format for the ecmContentBlock function:

```
<%= ecmContentBlock (int id) %>
```

Here is the format for the ecmContentBlockEX function:

```
<%= ecmContentBlockEX (int id, String xslt, int OverrideId) %>
```

These function is basically the same as the Content Block functions in ASP. See ["Content Block Function" on page 609](#) for details.

The only difference between the JSP function and the ASP one is the examples of implementing it. The JSP examples are below.

## Static Content Block

```
<tr>
 <td>
 <%= ecmContentBlock(1) %>
 </td>
</tr>
```

## Dynamic Content Block

```
<tr>
 <td>
 <%= ecmContentBlock($id) %>
 </td>
</tr>
```

## XML Content Block

```
<tr>
 <td>
 <%= ecmContentBlockEx (13, "", 1) %>
 </td>
</tr>
```

## Form Function

The `ecmFormBlock` function displays an HTML form content block. This function has a parameter that lets you decide whether to have form tags added automatically or manually. The following is an example of form tags.

```
<form name="myform" action="#" onsubmit="return
EkFmValidate(this);" method="post">

</form>
```

For more information about HTML forms, see the **Ektron CMS400.NET Administrator** manual chapter “Working with HTML Forms.”

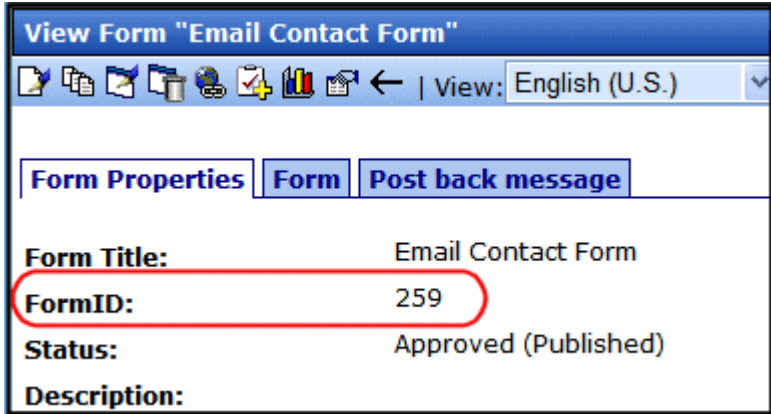
### IMPORTANT!

If you create a template for an existing form content block, you must manually change its quicklink to point to the new template. This change does not occur automatically. This procedure is described in the **Ektron CMS400.NET Administrator** manual chapter “Managing Library Assets”, section “Updating Default Template for Multiple Quicklinks.”

Here is the format for the `ecmFormBlock` function:

```
<%= ecmFormBlock (int id, boolean formTagReqd) %>
```

The ecmFormBlock function attributes are explained below.

Attribute	Description
id	<p>The "id" argument represents the form's ID number, visible when viewing the form from the View Form screen.</p> 
formTagReqd	<p>Allows form tags to be added automatically.</p> <p>This is an example of form tags.</p> <pre>&lt;form name="myform" action="#" onsubmit="return EkFmValidate(this);" method="post"&gt; &lt;/form&gt;</pre> <p><b>true</b> = Add form tags automatically  <b>false</b> = Add form tags manually</p>

This function is basically the same as the Form function in ASP. See ["Form Function" on page 678](#) for details.

The only differences between the JSP function and the ASP one are the formTagReqd parameter and the syntax for implementing it. The JSP example is below.

```
<tr>
 <td>
 <%= ecmFormBlock(1, true) %>
 </td>
</tr>
```

## Language Select Function

This function lets a visitor to your Web site select a language in which to view the site. Here is what the function looks like when published on a Web page.



Shown here is the format of the ecmLanguageSelect function.



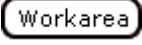
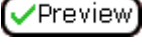
```
<%= ecmLanguageSelect() %>
```


This function is basically the same as the Show Site Language function in ASP. See ["Show Site Language Function" on page 687](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it.

## Login Function

This function is responsible for displaying the following buttons on the Web page.

Button	Name	Description
	Login	When user is not logged into Ektron CMS400.NET, this button appears. Click this button to let the user log in to the application.
	Logout	Once logged in, this button replaces the login button to allow the user to log out from the Ektron CMS400.NET Web site.
	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.

Button	Name	Description
	Preview Off	Turns off preview mode.

Shown here is the format of the ecmLogin function.

```
<%= ecmlogin() %>
```

This function is basically the same as the Login function in ASP. See "[Login Function](#)" on page 607 for details. The only difference is the syntax used to implement the function. This is listed below.

## Login Function Example

```
<tr>
 <td>
 <%= ecmlogin() %>
 </td>
</tr>
```

## DHTML Menus

**Ektron CMS400.NET**'s Menu feature lets users create and maintain a dropdown menu system for your Web site. The Menu feature lets you display menu options that link to content blocks, library assets, external hyperlinks and submenus.

To learn more about menus, please refer to the "Menu Feature" chapter of the **Ektron CMS400.NET** User Manual.

## Additional Documentation in Ektron Knowledge Base

This chapter explains how to implement standard menus that ship with **Ektron CMS400.NET**. Ektron's Web site also provides Knowledge Base articles that explain how to extend the standard menu features.

## Displaying a Menu on a Web Page

### *GenerateMenu and DropMenu*

**IMPORTANT!** The menu functions described below are only one example of how to implement menus. You can implement your own menu system using JavaScript.

Use the `GenerateNextGenMenu` and `DropNextGenMenu` custom functions to display a DHTML menu on a Web page. Place `GenerateNextGenMenu` once on the page to provide an infrastructure for `DropNextGenMenu`. Then, place `DropNextGenMenu` as needed to implement individual menus.

Follow these steps to display a menu on a Web template.



1. Place the following code directly after the page's <body> tag.

```
<%= ecmDHTML_GenerateNextGenMenu("menu id number") %>
```

For example:

```
<%= ecmDHTML_GenerateNextGenMenu("1") %>
```

To display more than one menu, separate each id number with a comma. For example:

```
<%= ecmDHTML_GenerateNextGenMenu("1,2,3") %>
```

2. If the user clicks *text* to display the menu, place the following code within the template file where the text would appear.

```
<%= ecmDHTML_DropNextGenMenu menu ID number,"text user clicks to display menu" %>
```

For example:

```
<%= ecmDHTML_DropNextGenMenu 1,"MyMenu1" %>
```

If the user clicks a *graphic* to invoke the menu, use this syntax:

```
<%= ecmDHTML_DropNextGenMenu menu ID number, '' %>
```

For example:

```
<%= ecmDHTML_DropNextGenMenu 1,'' %>
```

## Customizing the Menu's Appearance

You can customize a menu's appearance, such as the font style, color, and size. So, you have the flexibility to coordinate a menu with the design your Web site.

To customize a menu in **Ektron CMS400.NET**, you can change the ek\_Menu.js file.

### *ek\_Menu.js*

You can change the look and feel of a menu by changing the ek\_menu.js file. If you used the default settings when installing **Ektron CMS400.NET**, this file is in *webroot/siteroot/Workarea/java/*. For a description of the ek\_menu.js file, see ["ek\\_Menu.js" on page 689](#).

## XML Menu Function

The ecmGetMenuXML function wraps XML around the properties of a menu created in the Ektron CMS400.NET Workarea. Once created you can use the XML to display the menu. The ecmGetMenuXML function allows you to easily define how the menu data appears on the Web page.

```
<%= ecmGetMenuXML(int id) %>
```

This function is basically the same as the XML Menu function in ASP. See ["XML Menu" on page 692](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

```
<%= ecmGetMenuXML(1) %>
```

## Metadata Function

The metadata function is used to display the metadata for one or more content blocks in the Ektron CMS400.NET Web site. Inserting this custom function allows you to generate the list of metadata that is edited by the authors, and insert it in the source of the Web page. Shown below is an example of what the metadata custom function looks like:

```
<%= ecmMetadata(String itemlist, java.lang.Object spare1) %>
```

This function is basically the same as the Metadata function in ASP. See ["Metadata Function" on page 681](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP examples are below.

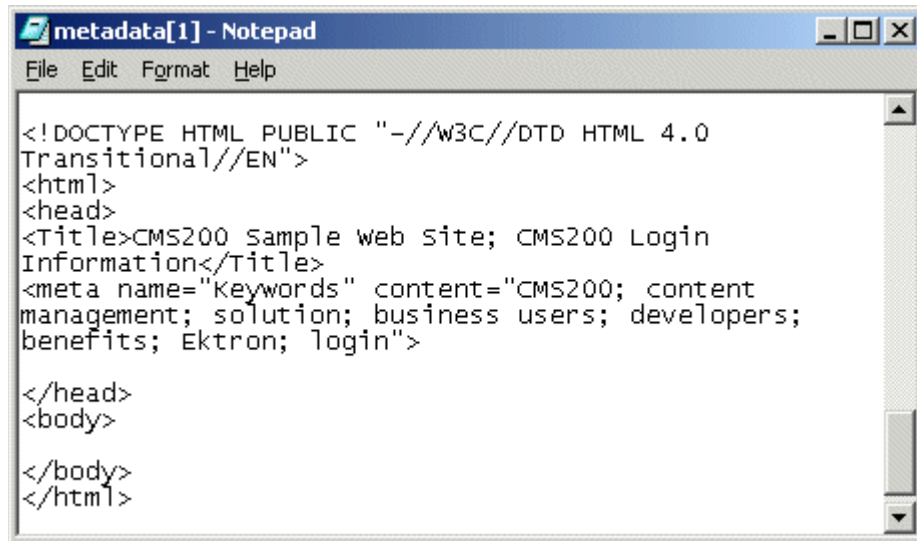
To include both content blocks' metadata in the web page, an ecmMetadata call would look like this.

```
<html>
 <head>
 <%= ecmMetadata ("[1][2;Title,Content-Type]", "") %>
 (rest of head)
 </head>
 <body>
 (rest of web page document)
 </body>
</html>
```

In the above example the ecmMetadata call is contained in the <head> of the HTML document (since all HTML metadata is required to exist there). Once the Web page is viewed the Web page source will contain the following information:

```
<html>
 <head>
 <Title>Welcome to Our Home Page</Title>
 <meta name="Keywords" content= Home; page; company; Welcome; Announce;
 product; widget">
 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
 (rest of head)
 </head>
 <body>
 (rest of web page document)
 </body>
</html>
```

```
<%= ecmMetaData ("[1][2]", "") %>
```



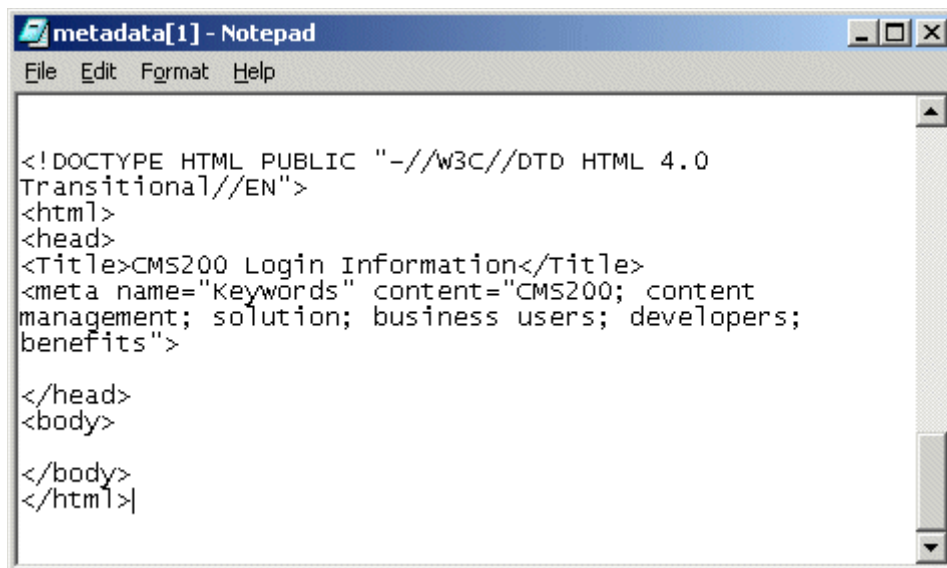
```
metadata[1] - Notepad
File Edit Format Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 sample web site; CMS200 Login
Information</Title>
<meta name="Keywords" content="CMS200; content
management; solution; business users; developers;
benefits; Ektron; login">

</head>
<body>

</body>
</html>
```

```
<%= ecmMetaData ("[1;title][2;keywords]", "") %>
```



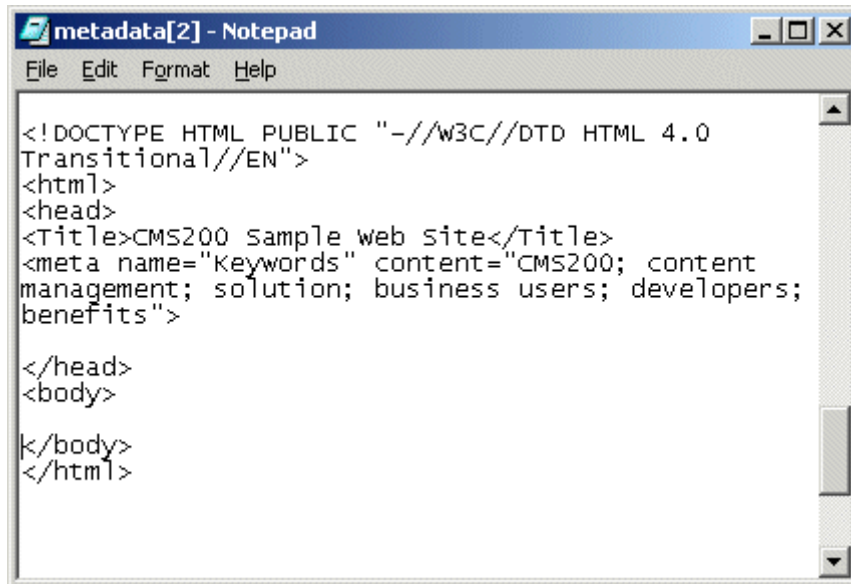
```
metadata[1] - Notepad
File Edit Format Help

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
<Title>CMS200 Login Information</Title>
<meta name="keywords" content="CMS200; content
management; solution; business users; developers;
benefits">

</head>
<body>

</body>
</html>
```

```
<%= ecmMetaData ("[1][2;title,keywords]", "") %>
```



## Search Function

The search function is used to display a text box with a search button on the template. When seen on the template on the Web page, a user can enter text and click the search button to search through all the content on the Web site.

Ektron has created two search functions:

- ecmSearch
- ecmSearchNoFormTag

Both functions have the same parameters and perform the same search. The only difference is ecmSearch automatically inserts form tags, ecmSearchNoFormTag does not.

The following is an example of form tags.

```
<form name="ecmsearch" method="post" action='search-
display.asp'>
```

```
</form>
```

When using ecmSearch, form tags are added automatically. When using ecmSearchNoFormTag you need to add the tags manually

Here is the format for the search function. You can interchange ecmSearch and ecmSearchNoFormTag when using this function.

```
<%= ecmSearch(java.lang.String startingFolder, boolean recursive, java.lang.String target, int
textBoxSize, int maxCharacters, java.lang.String buttonImageScr, java.lang.String buttonText,
java.lang.String fontFace, java.lang.String fontColor, java.lang.String fontSize, boolean
horizontal, java.lang.Object spare1) %>
```

This function is basically the same as the Search function in ASP. See ["Search Functions" on page 621](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

### Search Function Example

```
<TR>
 <TD>
 <%= ecmSearch("\\",true,"search-
display.jsp",25,200,"","Search","Verdana","#808080","2",false,"") %>
 </TD>
</TR>
```

## Search Display Function

The search display custom function creates the area where all the search results are shown after the search has been executed.

Here is the format of the ecmSearchDisplay function:

```
<%= ecmSearchDisplay(int maxNumber, String styleInfo, boolean ShowDate,
boolean showSummary) %>
```

This function is basically the same as the Search Display function in ASP. See ["Search Display Function" on page 631](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

### Search Display Function Example

```
<tr>
 <td>
 <%= ecmSearchDisplay(0,"",true,true) %>
 </td>
</tr>
```

## List Summary Function

The list summary function is used to display a list of content summaries on a Web page. Here is the format of the list summary function.

```
<%= ecmListSummary (String folder, boolean recursive, boolean showSummary,
String styleInfo, String OrderBy, String showInfo, int maxNumber, String objType,
String optionList, String summaryType) %>
```

This function is basically the same as the List Summary function in ASP. See ["List Summary Function" on page 635](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

### List Summary Example

```
<TR>
 <TD>
 <%= ecmListSummary ("Marketing\News",false,true,"",
 "Title,asc","DateModified",0,"","") %>
 </TD>
</TR>
```

## XML List Summary Function

The XML List Summary function wraps XML around the properties of a list of content, created as a list summary in the Ektron CMS400.NET Workarea. Once created, you can use the XML to display the list summary information. The ecmListSummaryXML function allows you to easily define how the list summary data appears on the Web page.

```
<%= ecmListSummaryXML (String Folder, int Recursive, String OrderBy, int MaxNumber,
String ObjType, String SummaryType) %>
```

This function is basically the same as the List Summary XML function in ASP. See ["XML List Summary Function" on page 642](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

```
<%= ecmListSummaryXML ("\\CMS400Min\\",0,"Title,asc",0,"folder","");
?>
```

## RSS List Summary Function

The RSS summary function is used to create an XML list of content summaries for RSS Syndication. Shown here is the format of the RSS list summary function:

```
<%= ecmRssSummary (java.lang.String Folder, boolean Recursive, java.lang.String OrderBy,
int MaxNumber) %>
```

This function is basically the same as the RSS Summary function in ASP. See ["RSS Summary Function" on page 652](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

```
<%= ecmRssSummary("\\CMS400Min\\", false, "Title,asc", 0) %>
```

## Single Summary Function

The single summary function is used to display the summary of one content block in the Ektron CMS400.NET Web site, instead of a list of all the summaries in a content folder. Seen below is the single summary function:

```
<%= ecmRssSummary (int contentID, boolean showSummary, java.lang.String showInfo,
java.lang.String styleInfo, java.lang.String spare1) %>
```

This function is basically the same as the Single Summary function in ASP. See ["Single Summary Function" on page 648](#) for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

## Single Summary Example

```
<TR>
 <TD>
 <%= ecmSingleSummary ("14",true,"","","") %>
 </TD>
</TR>
```

# Ektron Markup Language

## Introduction

The Ektron Markup Language (EkML) makes it easy for web developers to manage the output presentation of server controls. EkML uses a simple markup that resembles HTML in syntax. The EkML is cached and .NET watches for file changes.

The MarkupLanguage property in the Ektron server controls and Ektron Dreamweaver functions is used to select the template file that contains the Ektron Markup Language. This language is made up of tags and variables that assign formatting information to a control or function when displayed on a Web page.

---

**IMPORTANT!**

If the markup template file is located in the same folder as the form that contains the server control, just type the file's name in the server control's MarkupLanguage property. For example, `mycollectionmarkup.ekml`. If the file is in another folder, enter the path relative to site root. For example, `\cms400Min\workarea\customfiles\markup\mycollectionmarkup.ekml`.

---

This chapter contains the following topics.

- "EkML Templates" on page 754
- "EkML Example" on page 755
- "EkML Tags" on page 758
- "EkML Variables" on page 758
- "collection.ekml" on page 778
- "contentlist.ekml" on page 781
- "listsummary.ekml" on page 783
- "map.ekml" on page 787
- "metadatalist.ekml" on page 791
- "taxonomy.ekml" on page 794
- "websearch.ekml" on page 798

## EkML Templates

Ektron, Inc. has provided an .ekml basic template for each control. These templates are located in [webroot]/Workarea/Templates. At the top of each template is a list of variables that can be used with that control.



**IMPORTANT!**

Ektron strongly recommends that you save the template under another name and use that template. This prevents your file from being overwritten when upgrades occur.

**NOTE**

If you install the CMS400Min site, .ekml template files reside in various project folders. These files exemplify using the .ekml with a specific server control.

The following server controls and Dreamweaver functions currently use the Ektron Markup Language:

- **Collection** - "collection.ekml" on page 778
- **ContentList** - "contentlist.ekml" on page 781
- **ListSummary** - "listsummary.ekml" on page 783
- **Map** - "map.ekml" on page 787
- **MetadataList** - "metadatalist.ekml" on page 791
- **Taxonomy/Directory** - "taxonomy.ekml" on page 794
- **WebSearch** - "websearch.ekml" on page 798

## EkML Example

By customizing these templates, you can create a custom layout for the content you are displaying. For example, if you have a Collection and you want to display it as a numbered list containing a content's hyperlink, the date it was last updated and its summary, you would create the following .ekml file and assign that file to a Collection server control's MarkupLanguage property.

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink] Updated: <i>[$DateModified]</i>
[$Teaser] <hr/>
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

Once the Collection is displayed, it is formatted as follows:

1. [Ektron Rated Positive](#) Updated: 2/28/2006 3:53:51 PM  
Ektron Inc., an innovator in Web content management software, today announced ...

---

2. [Ektron to Demonstrate Healthcare](#) Updated: 2/28/2006 3:54:07 PM  
"Healthcare Content Indexing Framework" which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.

Because you can use HTML in the Ektron Markup Language, you can format the variables using common HTML tags.

Let's breakdown the example collection.ekml code found above.

- The `<ekmarkup>` tags open and close the markup language. Everything to do with the EkML needs to be between these tags.
- The specific information you want displayed and any HTML formatting are added between the `<ekoutput>` tags.
- `<table width="100%" border="0"></table>` sets up a table. This is HTML formatting.
- The `<ekrepeat>` tags contain formatting information and variables for items in the Collection. It repeats this information for each item in the list.
- The `[$Index]` variable creates a numbered list for each content item in the Collection. Note, you can add a period (.) or other separator depending on how you want the list to look.

1. [Ektron Rated Positive](#)  
Ektron Inc., an innovato  
announced ...

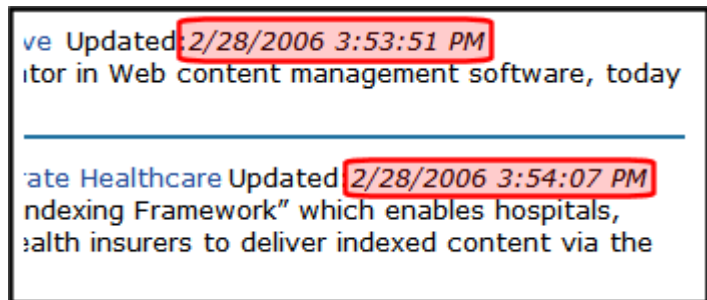
---

2. [Ektron to Demonstrat](#)  
"Healthcare Content Ind  
medical groups and heal  
Web.

- The `[$HyperLink]` variable adds a hyperlink for each content item in the Collection. The hyperlink use the content's title as the text for the link. There is no need for anchor tags, the variable creates them for you. If you want to use anchor tags for formatting you own links, you can use the `[$QuickLink]` variable.



- Updated: is plain text.
- The <i></i> tags are HTML that cause the [\$DateModified] variable to appear in italics.
- The [\$DateModified] variable displays the date and time each content item in the Collection was last modified.



- <br/> is HTML. It adds a line break.
- The [\$Teaser] variable displays the summary for each content item in the Collection.



- <hr/> is HTML. It adds a horizontal rule line.

## EkML Tags

The EkML tags define functions that occur when using the language to display content. The Ektron Markup Language includes the following tags.

EkML Tag	Description
<ekmarkup>	Use these tags to open and close the Ektron Markup Language.
<ekoutput>	Define what information is output from the server control between these tags.
<ekrepeat>	These tags cause what ever formatting information that appears between them to be applied to each item in the list. They always appear between the <ekoutput> tags.
<ekbubbleinfo>	Creates a pop-up bubble. This tag is invoked when you use the [ <code>\$ShowBubble</code> ] or [ <code>\$ShowBubble(width)</code> ] variables. By adding different variables between the <ekbubbleinfo> tags, you define what information appears in the bubble. These tags are placed outside the <ekoutput> tags but within the <ekmarkup> tags.
<ekcontentinfo>	Places the information defined between the tags in the specified HTML tag ID. This tag is invoked when you use the [ <code>\$ShowContent('htmltagid')</code> ] variable.

## EkML Variables


The Ektron Markup Language uses variables that appear between the tags in a template file to define the information that appears in a control's display. Some variable are used in more than one template. These are known as common EkML variables.

Another type of variable is a server control specific variable, these variables can only be used with a specific server control. For example, [`$SearchSummary`] can only be used in WebSearch.

### EkML Variables

Below is a list of EkML variables with a short description, a link to additional information and whether the variable is common or specific.

Variable	Description	Common or Control Specific	More Information
[\$CollectionDescription]	Display the collection's description.	Collection	"[\$CollectionDescription]" on page 780
[\$CollectionTitle]	Display the collection's title.	Collection	"[\$CollectionTitle]" on page 781
[\$Comment]	The content's comment information is displayed.	Common	"[\$Comment]" on page 763
[\$ContentByteSize]	Displays the content item's size in KB. Works only with taxonomy search. <code>&lt;ekoutput mode="article_search"&gt;</code>	Directory WebSearch	"[\$ContentByteSize]" on page 801
[\$ContentId]	Displays the content item's ID.	Common	"[\$ContentId]" on page 763
[\$DateCreated]	Display the date and time the content was created.	Common	"[\$DateCreated]" on page 764
[\$DateModified]	Display the date and time the content was last modified.	Common	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	Common	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	Common	"[\$EditorLastName]" on page 766
[\$FolderDescription]	Displays the folder's description.	ListSummary	"[\$FolderDescription]" on page 786
[\$FolderId]	Display the folder ID of a content item.	Common	"[\$FolderId]" on page 766
[\$FolderName]	Displays the folder's name.	ListSummary	"[\$FolderName]" on page 787

Variable	Description	Common or Control Specific	More Information
[\$Html]	Display the HTML contained in the content item.	Common	"[\$Html]" on page 766
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	Common	"[\$HyperLink]" on page 767
[\$Image]	Displays the path to the image defined in a content item's <b>Metadata &gt; Image data</b> field. When wrapped in <code>&lt;img src="" /&gt;</code> tag, the image is displayed.	Common	"[\$Image]" on page 768
[\$ImageIcon]	Displays an image icon for the content item. For example, if the content item is HTML, the (  ) icon is displayed.	Common	"[\$ImageIcon]" on page 768
[\$ImageThumbnail]	Displays the path for the image's thumbnail defined in a content item's Metadata.	Common	"[\$ImageThumbnail]" on page 769
[\$Index]	Serialize the content items in a numbered list.	Common	"[\$Index]" on page 769
[\$ItemCount]	The total number of items in a list.	Common	"[\$ItemCount]" on page 769
[\$Language]	Display the language ID for the content item.	Common	"[\$Language]" on page 769
[\$LinkTarget]	When added to an <code>&lt;a href=""&gt;</code> tag's <code>target=""</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	Common	"[\$LinkTarget]" on page 770
[\$PagingCurrentEndIndex]	The end count number of the items on the page. For example, if you are displaying items 11 - 20 on a page, this variable represents the number 20. Works only with taxonomy search. <code>&lt;ekoutput mode="article_search"&gt;</code>	Directory & WebSearch	"[\$PagingCurrentEndIndex]" on page 801

Variable	Description	Common or Control Specific	More Information
[\$PagingCurrentStartIndex]	The numerical record of the first item on a page. For example, if you are displaying items 1 through 10 out of 50 total items on a page, this variable represents the number 1. Works only with taxonomy search. <ekoutput mode="article_search">	Directory & WebSearch	"[\$PagingCurrentStartIndex]" on page 803
[\$QuickLink]	This property displays the Quicklink information for the content item. When wrapped in an <a href=""> tag, you can create a Hyperlink.	Common	"[\$QuickLink]" on page 770
[\$SearchDuration]	Displays the amount of time, in seconds, it has taken to execute the search. Works only with taxonomy search. <ekoutput mode="article_search">	Directory & WebSearch	"[\$SearchDuration]" on page 804
[\$SearchSummary]	Creates a summary from information stored in the indexing service for each item in the search results. Works only with taxonomy search. <ekoutput mode="article_search">	Directory & WebSearch	"[\$SearchSummary]" on page 800
[\$SearchText]	Displays the text for which a user is searching. This information is same as what a user entered in the search text box. Works only with taxonomy search. <ekoutput mode="article_search">	Directory & WebSearch	"[\$SearchText]" on page 803
[\$SERVER_NAME]	Displays the server name. For example, If this variable is applied to http://www.example.com/demo.aspx, the return is www.example.com.	Common	"[\$SERVER_NAME]" on page 771
[\$ShowAllcategory]	Adds to the Taxonomy search screen a checkbox that lets the user decide if he wants to display categories that have no items.	Directory	"[\$ShowAllcategory]" on page 771

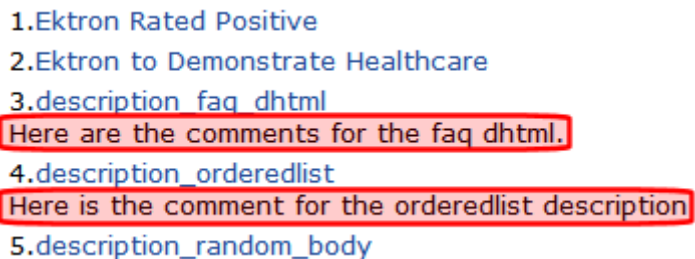
Variable	Description	Common or Control Specific	More Information
<code>[\$ShowBubble(width)]</code>	This is similar to <code>[\$ShowBubble]</code> . It calls the <code>&lt;ekbubbleinfo&gt;</code> tags and allows you to set the width and height of the bubble. For example, <code>[\$ShowBubble(300,400)]</code> . The first number represents the width. The second number represents the height.	Common	" <code>[\$ShowBubble(width)]</code> " on page 773
<code>[\$ShowBubble]</code>	Calls the <code>&lt;ekbubbleinfo&gt;</code> tags and places the information contained within those tags in a pop-up bubble.	Common	" <code>[\$ShowBubble]</code> " on page 772
<code>[\$ShowContent('htmltagid')]</code>	Calls the <code>&lt;ekcontentinfo&gt;</code> tags and places the information in those tags within the specified HTML tag ID. Replace the <code>htmltagid</code> with the ID of the tag.	Common	" <code>[\$ShowContent('htmltagid')]</code> " on page 773
<code>[\$Status]</code>	Displays the status of a content item.	Common	" <code>[\$Status]</code> " on page 774
<code>[\$Teaser]</code>	Display the content item's summary information.	Common	" <code>[\$Teaser]</code> " on page 774
<code>[\$TemplateQuickLink]</code>	This property displays the Template Quicklink information assigned to the taxonomy item in the Workarea. When wrapped in an <code>&lt;a href=""&gt;</code> tag, you can create a Hyperlink.  See Also: Ektron CMS400.NET Administrator Manual section "Taxonomy" > "Creating a Taxonomy in the Workarea" > "Applying a Template to a Taxonomy or Category"	Directory	" <code>[\$TemplateQuickLink]</code> " on page 798
<code>[\$Title]</code>	Displays the content item's title.	Common	" <code>[\$Title]</code> " on page 775



Variable	Description	Common or Control Specific	More Information
<code>[\$UrlEncode('str')]</code>	Encodes the string information. Replace <b>str</b> with the string you want to encode. This variable can be used to encode another EkML variable and place it in an email.	Common	" <code>[\$UrlEncode('str')]</code> " on page 777
<code>[\$UrlParam('paramname')]</code>	Displays the value of a query string's given parameter. For example, if the query string is <code>?id=27</code> and the variable is <code>[\$UrlParam('id')]</code> , 27 is displayed.	Common	" <code>[\$UrlParam('paramname')]</code> " on page 776

**`[$Comment]`**

The content's comment information is displayed. Comment information can be added in the Workarea via the content item's comment tab.



```

1.Ektron Rated Positive
2.Ektron to Demonstrate Healthcare
3.description_faq_dhtml
Here are the comments for the faq dhtml.
4.description_orderedlist
Here is the comment for the orderedlist description
5.description_random_body

```

```

<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink]
 [$Comment]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>

```

**`[$ContentId]`**

Displays the content item's ID.

1. Bubble Example Content ID = 748
2. Ektron Rated Positive Content ID = 32
3. Ektron to Demonstrate Healthcare Content ID = 31
4. description\_faq\_dhtml Content ID = 182
5. description\_orderedlist Content ID = 184

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink] Content ID = [$ContentId]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

### ***[ \$DateCreated ]***

Display the date and time the content was created.

1. Ektron Rated Positive - 2/14/2006 9:30:07 AM
2. Ektron to Demonstrate Healthcare - 2/14/2006 9:17:04 AM
3. description\_faq\_dhtml - 2/27/2006 9:40:37 AM
4. description\_orderedlist - 2/27/2006 9:42:23 AM
5. description\_random\_body - 2/27/2006 9:38:37 AM
6. description\_random\_teaser - 2/27/2006 9:36:55 AM

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink] - <i>[$DateCreated]</i>
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

**[\$DateModified]**

Display the date and time the content was last modified.

1.Ektron Rated Positive Date Modified: <b>2/28/2006 3:53:51 PM</b> 2.Ektron to Demonstrate Healthcare Date Modified: <b>2/28/2006 3:54:07 PM</b> 3.description_faq_dhtml Date Modified: <b>11/30/2006 4:13:34 PM</b> 4.description_orderedlist Date Modified: <b>11/30/2006 4:18:21 PM</b>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink]
Date Modified: [$DateModified]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

**[\$EditorFirstName]**

Display the last editor's first name for a content item.

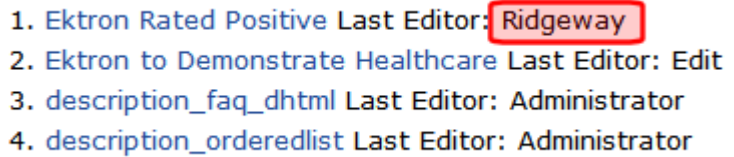
1. Ektron Rated Positive Last Editor: <b>Frank</b> 2. Ektron to Demonstrate Healthcare Last Editor: John 3. description_faq_dhtml Last Editor: Application 4. description_orderedlist Last Editor: Application
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink] Last Editor: [$EditorFirstName]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
```

</ekmarkup>

### **[\$EditorLastName]**

Display the last editor's last name for a content item.

- 
1. Ektron Rated Positive Last Editor: Ridgeway
  2. Ektron to Demonstrate Healthcare Last Editor: Edit
  3. description\_faq\_dhtml Last Editor: Administrator
  4. description\_orderedlist Last Editor: Administrator

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$Index]. [$HyperLink] Last Editor: [$EditorLastName]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

### **[\$FolderId]**

Display the folder ID of a content item.

### **[\$Html]**

Display the HTML contained in the content item. In the server control, the `GetHTML` property must be set to **True**; otherwise nothing is displayed. The exception to this is when the `[$Html]` appears between the `<ekbubbleinfo>` tags. In that case, the `GetHTML` property can be set to **True** or **False**.

## Ektron Rated Positive by Leading Analyst Firm

```
<ekmarkup>
 <ekoutput>
 <table width="100%" border="0">
 <ekrepeat>
 <tr>
 <td>
 [$HyperLink]
[$Html]
 </td>
 </tr>
 </ekrepeat>
 </table>
 </ekoutput>
</ekmarkup>
```

Adds a hyperlink using the title of the content block as the text. You do not add an `<a href="">` tag when using this variable. That functionality is built into the EkML. Use this variable when you do not want to create a custom hyperlink. If you want to create a custom hyperlink, use the `[QuickLink]` variable. See Also: "`[QuickLink]`" on page 770

- ```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
```

```

<ekrepeat>
  <tr>
    <td>
      [$Index]. [$HyperLink]
    </td>
  </tr>
</ekrepeat>
</table>
</ekoutput>
</ekmarkup>



```

[\$Image]

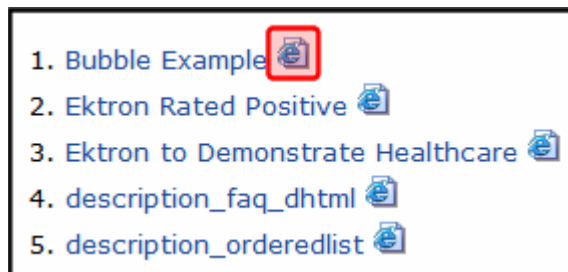
Displays the path for the image defined in a content item's Metadata. When wrapped in `` tag, the image is displayed. For example, ``.

An example of using this variable is on the developer sample page > List Summary > Flash news. That sample (flashnews.aspx) refers to flashnews.ekml, which illustrates using this variable.

[\$ImageIcon]

Displays an image icon for the content item. Except for HTML content, these icons are the same icons used in the Workarea to show the content type. HTML content uses the Internet Explorer icon (). For example, Forms use the Form icon ().

In the server control, the `IncludeIcons` property must be set to **True**; otherwise nothing is displayed.



```

<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$HyperLink] [$ImageIcon]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>

```

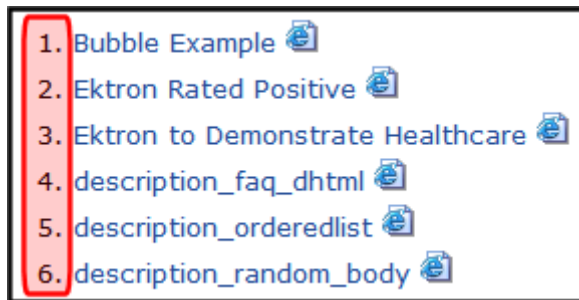
[\$ImageThumbnail]

Displays the path for the image's thumbnail defined in a content item's Metadata. When wrapped in `` tag, the image thumbnail is displayed. For example, ``.

An example of using this variable is on the developer sample page > List Summary > Flash news. That sample (flashnews.aspx) refers to flashnews.ekml, which illustrates using this variable.

[\$Index]

Serialize the content items in a numbered list.



```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$HyperLink] [$ImageIcon]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

[\$ItemCount]

The total number of items in a list. In the server control, the EnablePaging property must be set to **True**; otherwise nothing is displayed

[\$Language]

Display the language ID for the content item.

[\$LinkTarget]

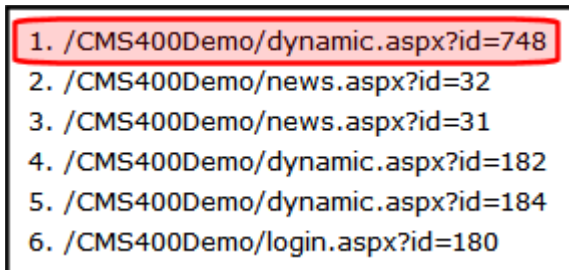
When added to an `` tag's `target=""` attribute, this variable reads the server control's `LinkTarget` property and uses its setting. For example, If you want to create a custom hyperlink that opens in a new window, you set the server control's `LinkTarget` property to `_Blank`. Then, in the Ektron Markup Language file, add the `[$LinkTarget]` variable to the `` tag's `Target=""` attribute. A code example appears below.

```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            <a href="[$QuickLink]" target="[$LinkTarget]">[$Title]</a>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

[\$QuickLink]

This property displays the Quicklink information for the content item. When wrapped in an `` tag, you can create a Hyperlink. Use this property instead of the `[$HyperLink]` if you want to customize your hyperlinks. See Also: "[\[\\$HyperLink\]](#)" on page 767

The first image shows the variable displaying the Quicklink information. The second image shows the Quicklink as a Hyperlink with a custom text.



```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$QuickLink]
          </td>
        </tr>
      </ekrepeat>
    </table>
```



```
</ekoutput>
</ekmarkup>
```



```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [ $Index ]. <a href="[ $QuickLink ]">Click Here For Item # [ $Index ]</a>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

[\$SERVER_NAME]

Displays the server name for the page on which this variable appears. For example, If this variable is applied to `http://www.example.com/demo.aspx`, the return is `www.example.com`.

[\$ShowAllcategory]

If a Directory server control's `EnableSearch` property is set to `true`, this variable adds the checkbox circled below to the right of the search field.

☒ the Directory ☐ this Category

Search:

> Top

Category: ([What's This?](#)) ☒ show all (include Categories with no items)

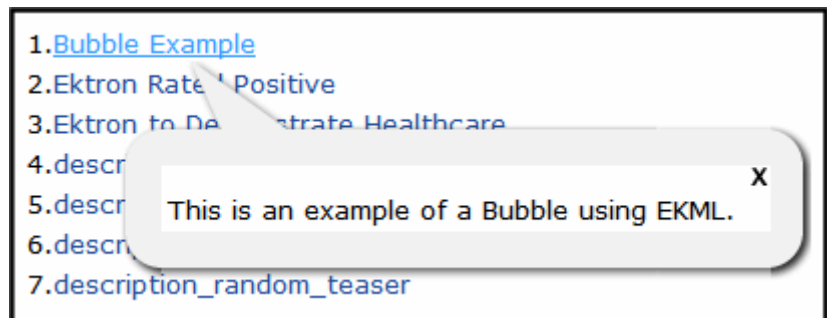
Articles: ([What's This?](#))

-  [LisaGlobalizationIndustryPrimer3_en](#)

By default, the Directory server control only shows categories to which at least one content block is assigned. If you check this box, all categories appear, even those with no content assigned.

[\$ShowBubble]

Calls the <ekbubbleinfo> tags and places the information contained within those tags in a pop-up bubble. This bubble is typically assigned to the `onclick` or `onmouseover` attribute in an tag. See the example EkML code below.



```

<ekmarkup>
  <ekbubbleinfo>
    <table border="0">
      <tr>
        <td>[$Html]</td>
      </tr>
    </table>
  </ekbubbleinfo>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index].<a href="#" onmouseover="[$ShowBubble]">[$Title]</a>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>

```

```

        </tr>
    </ekrepeat>
</table>
</ekoutput>
</ekmarkup>

```

[`$ShowBubble(width)`]

This is similar to [`$ShowBubble`]. It calls the `<ekbubbleinfo>` tags and places the information contained within those tags in a pop-up bubble. This variable allows you to set the width and height of the bubble. For example, [`$ShowBubble(300,400)`]. In this example, the first number represents the width. The second number represents the height.

If you enter a single number, it sets the width. The height of the bubble is then limited to the length of the content. For example, if you had a video that was formatted at 200 pixels wide and you wanted to launch it in a bubble, you would add the function as [`$ShowBubble(200)`].

The minimum width for a bubble is 287. The minimum height is 101.

```

<ekmarkup>
  <ekbubbleinfo>
    <table border="0">
      <tr>
        <td>[$Html]</td>
      </tr>
    </table>
  </ekbubbleinfo>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index].<a href="#" onmouseover="$ShowBubble(200)">[$Title]</a>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>

```

[`$ShowContent('htmltagid')`]

Calls the `<ekcontentinfo>` tags and places the information contained within those tags in the specified HTML tag ID. Replace the `htmltagid` with the ID of the tag. This tag is typically assigned to the `onclick` or `onmouseover` attribute in an `` tag. See the example EkML code below.

```

<ekmarkup>
  <ekcontentinfo>
    <table border="0">
      <tr>
        <td>[$Html]</td>
      </tr>
    </table>

```

```

</ekcontentinfo>
<ekoutput>
  <table width="100%" border="0">
    <ekrepeat>
      <tr>
        <td>
          [ $Index ]. <a href="#" onmouseover="[$ShowContent('contarea')]">[ $Title]</a>
        </td>
      </tr>
    </ekrepeat>
  </table>
</ekoutput>
</ekmarkup>

```

[\$Status]

Displays the status of a content item. For example, Approved, Checked in, or Submitted for Approval. For additional information on content statuses, see the User Manual > "Appendix A: Content Statuses"



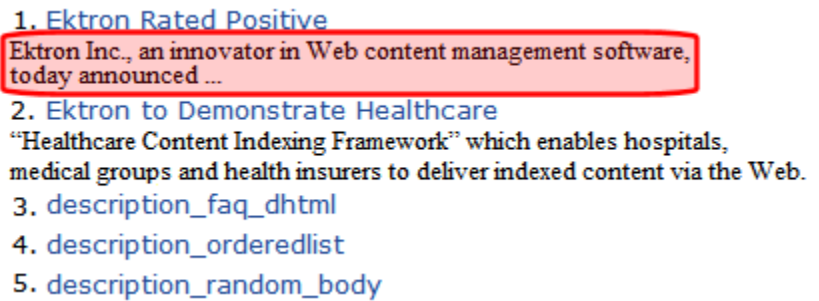
```

<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [ $Index ]. [ $Hyperlink ]
            The content status is <i>[ $Status ]</i>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>

```

[\$Teaser]

Display the content item's summary information. Summary information is added to content in the Workarea via the Summary tab. A code example appears below.



1. Ektron Rated Positive
 Ektron Inc., an innovator in Web content management software, today announced ...

2. Ektron to Demonstrate Healthcare
 "Healthcare Content Indexing Framework" which enables hospitals, medical groups and health insurers to deliver indexed content via the Web.

3. description_faq_dhtml

4. description_orderedlist

5. description_random_body

```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [ $Index ]. [ $HyperLink ] <br/> [ $Teaser ]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

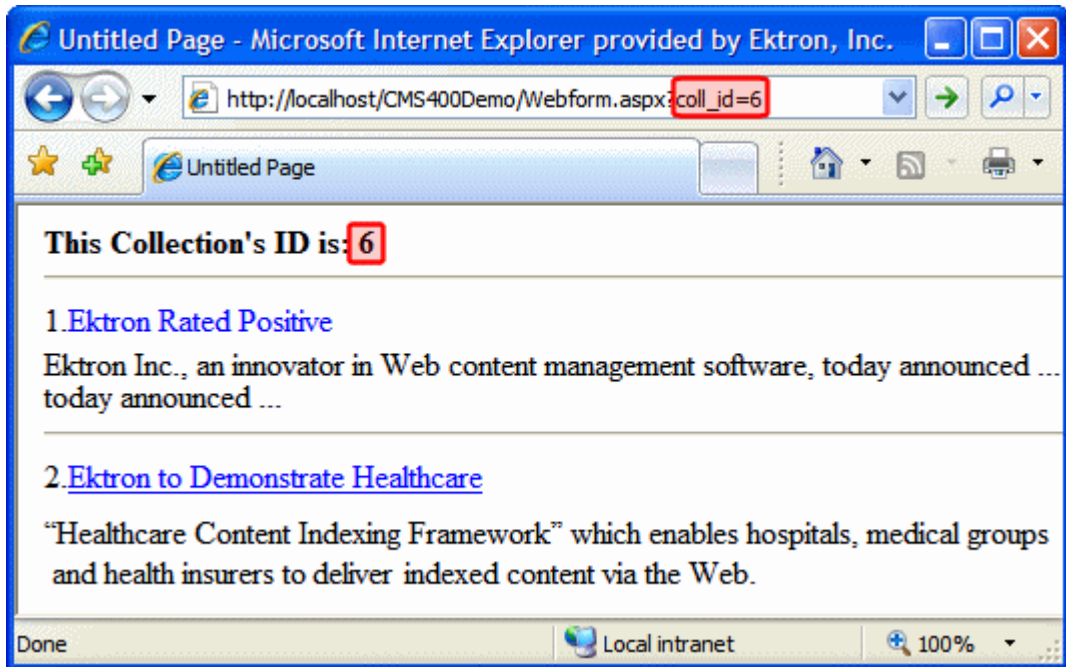
[\$Title]

Displays the content item's title. Use this variable if you want to display the title as normal text. If you want to display the title as a hyperlink, use [\$HyperLink]. See Also: "[\$HyperLink]" on page 767. If you want to create a custom hyperlink with the title as the text of the hyperlink, use this property in conjunction with the tag and the [\$QuickLink] variable. See Also: "[\$QuickLink]" on page 770.

```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [ $Index ]. <a href="[ $QuickLink ]">[ $Title ]</a>
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

[\$UrlParam('paramname')]

Displays the value of a query string's parameter. For example, if you have a collection and want to display its ID, add the [\$UrlParam('coll_id')] where you want the collection's ID to appear.



```
<ekmarkup>
  <ekbubbleinfo>
    <table border="0">
      <tr>
        <td>[$Html]</td>
      </tr>
    </table>
  </ekbubbleinfo>
  <ekoutput>
    <table width="100%" border="0">
      <tr>
        <td>
          <b>This Collection's ID is: [$UrlParam('coll_id')]</b>
          <br/><hr/>
        </td>
      </tr>
    </table>
    <ekrepeat>
      <tr>
        <td>
          [$Index].<a href="#" onclick="[$ShowBubble]">[$Title]</a>[$Teaser]<hr/>
        </td>
      </tr>
    </ekrepeat>
  </ekoutput>
</ekmarkup>
```

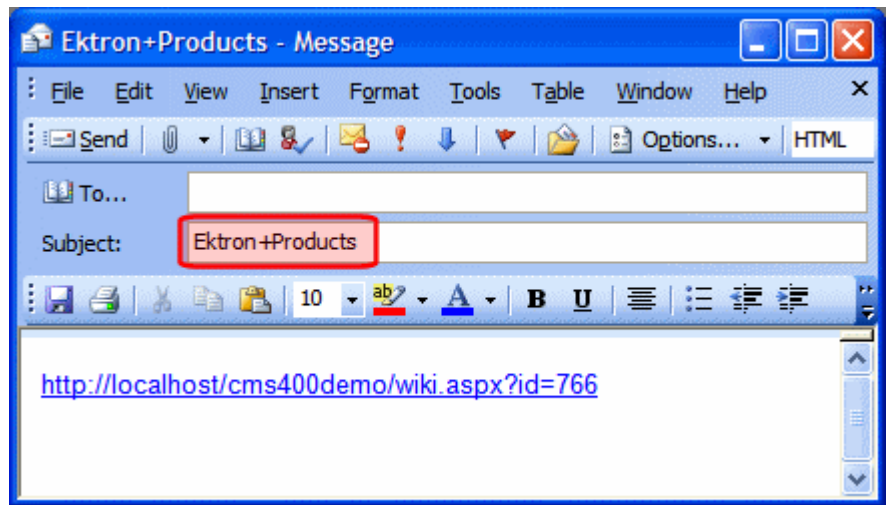
```

    </ekrepeat>
  </table>
</ekoutput>
</ekmarkup>

```

[\$UrlEncode('str')]

Encodes the string information in the variable. This variable can be used to encode another EkML variable and place it in an email. For example, You want the title of the content block to appear as the subject of an email. A code example appears below.



```

<ekcontentinfo>
  <table border="0">
    <tr>
      <td>
        <strong>Title:&#160;</strong>
        <span style="color:red;">[$Title]</span>
        <a href="mailto:?subject=[$UrlEncode('[$Title]')]&body=http://[$SERVER_NAME]/
CMS400Min/wiki.aspx?id=[$ContentId]">&nbsp;</a>
        <hr />
      </td>
    </tr>
    <tr>
      <td>[$Html]</td>
    </tr>
  </table>
</ekcontentinfo>

```

EkML Template Examples


collection.ekml

This file defines which items and information are included when displaying content item information in a collection using the Collection server control. For an example of using this file with a Collection server control, see the CMS400 demo example: http://your_site/siteroot/Developer/Collection/TemplateMarkup.aspx.

collection.ekml Variables

Below is a list of variables that are used with the collection.ekml file.

Variable	Description	More Information
[\$CollectionDescription]	Display the collection's description.	"[\$CollectionDescription]" on page 780
[\$CollectionTitle]	Display the collection's title.	"[\$CollectionTitle]" on page 781
[\$Comment]	The content's comment information is displayed.	"[\$Comment]" on page 763
[\$ContentId]	Displays the content item's ID.	"[\$ContentId]" on page 763
[\$DateCreated]	Display the date and time the content was created.	"[\$DateCreated]" on page 764
[\$DateModified]	Display the date and time the content was last modified.	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	"[\$EditorLastName]" on page 766
[\$FolderId]	Display the folder ID of a content item.	"[\$FolderId]" on page 766
[\$Html]	Display the HTML contained in the content item.	"[\$Html]" on page 766
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	"[\$HyperLink]" on page 767

Variable	Description	More Information
<code>[\$Image]</code>	Displays the path for the image defined in a content item's Metadata. When wrapped in <code></code> tag, the image is displayed.	" <code>[\$Image]</code> " on page 768
<code>[\$ImageIcon]</code>	Displays an image icon for the content item. For example, if the content item is HTML, the  icon is displayed.	" <code>[\$ImageIcon]</code> " on page 768
<code>[\$ImageThumbnail]</code>	Displays the path for the image's thumbnail defined in a content item's Metadata.	" <code>[\$ImageThumbnail]</code> " on page 769
<code>[\$Index]</code>	Serialize the content items in a numbered list.	" <code>[\$Index]</code> " on page 769
<code>[\$ItemCount]</code>	The total number of items in a list.	" <code>[\$ItemCount]</code> " on page 769
<code>[\$Language]</code>	Display the language ID for the content item.	" <code>[\$Language]</code> " on page 769
<code>[\$LinkTarget]</code>	When added to an <code></code> tag's <code>target=""</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	" <code>[\$LinkTarget]</code> " on page 770
<code>[\$QuickLink]</code>	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	" <code>[\$QuickLink]</code> " on page 770
<code>[\$SERVER_NAME]</code>	Displays the server name. For example, If this variable is applied to <code>http://www.example.com/demo.aspx</code> , the return is <code>www.example.com</code> .	" <code>[\$SERVER_NAME]</code> " on page 771
<code>[\$ShowBubble(width)]</code>	This is similar to <code>[\$ShowBubble]</code> . It calls the <code><ekbubbleinfo></code> tags and allows you to set the width of the bubble.	" <code>[\$ShowBubble(width)]</code> " on page 773
<code>[\$ShowBubble]</code>	Calls the <code><ekbubbleinfo></code> tags and places the information contained within those tags in a pop-up bubble.	" <code>[\$ShowBubble]</code> " on page 772
<code>[\$ShowContent('htmltagid')]</code>	Calls the <code><ekcontentinfo></code> tags and places the information in those tags within the specified HTML tag ID. Replace the 'htmltagid' with the ID of the tag.	" <code>[\$ShowContent('htmltagid')]</code> " on page 773

Variable	Description	More Information
<code>[\$Status]</code>	Displays the status of a content item.	" <code>[\$Status]</code> " on page 774
<code>[\$Teaser]</code>	Display the content item's summary information.	" <code>[\$Teaser]</code> " on page 774
<code>[\$Title]</code>	Displays the content item's title.	" <code>[\$Title]</code> " on page 775
<code>[\$UrlEncode('str')]</code>	Encodes the string information. This variable can be used to encode another EkML variable and place it in an email.	" <code>[\$UrlEncode('str')]</code> " on page 777
<code>[\$UrlParam('paramname')]</code>	Displays the value of a query string's parameter. For example, if the query string is <code>?id=27</code> and the variable is <code>[\$UrlParam('id')]</code> , 27 is displayed.	" <code>[\$UrlParam('paramname')]</code> " on page 776

`[$CollectionDescription]`

Display the collection's description.



```

<ekmarkup>
  <ekoutput>
    <h3><b>[$CollectionTitle]</b></h3><p>[$CollectionDescription]<br/>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$HyperLink]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>

```

[\$CollectionTitle]

Display the collection's title.



```
<ekmarkup>
  <ekoutput>
    <h3><b>[$CollectionTitle]</b></h3>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$HyperLink]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

contentlist.ekml

This file defines which items and information are included when displaying content item information in a content list using the ContentList server control.

contentlist.ekml Variables

Below is a list of variables that are used with the contentlist.ekml file.

Variable	Description	More Information
[\$Comment]	The content's comment information is displayed.	"[\$Comment]" on page 763
[\$ContentId]	Displays the content item's ID.	"[\$ContentId]" on page 763
[\$DateCreated]	Display the date and time the content was created.	"[\$DateCreated]" on page 764

Variable	Description	More Information
[\$DateModified]	Display the date and time the content was last modified.	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	"[\$EditorLastName]" on page 766
[\$FolderId]	Display the folder ID of a content item.	"[\$FolderId]" on page 766
[\$Html]	Display the HTML contained in the content item.	"[\$Html]" on page 766
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	"[\$HyperLink]" on page 767
[\$Image]	Displays the path for the image defined in a content item's Metadata. When wrapped in <code></code> tag, the image is displayed.	"[\$Image]" on page 768
[\$ImageIcon]	Displays an image icon for the content item. For example, if the content item is HTML, the () icon is displayed.	"[\$ImageIcon]" on page 768
[\$ImageThumbnail]	Displays the path for the image's thumbnail defined in a content item's Metadata.	"[\$ImageThumbnail]" on page 769
[\$Index]	Serialize the content items in a numbered list.	"[\$Index]" on page 769
[\$ItemCount]	The total number of items in a list.	"[\$ItemCount]" on page 769
[\$Language]	Display the language ID for the content item.	"[\$Language]" on page 769
[\$LinkTarget]	When added to an <code></code> tag's <code>target=" "</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	"[\$LinkTarget]" on page 770
[\$QuickLink]	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	"[\$QuickLink]" on page 770

Variable	Description	More Information
<code>[\$SERVER_NAME]</code>	Displays the server name. For example, If this variable is applied to <code>http://www.example.com/demo.aspx</code> , the return is <code>www.example.com</code> .	" <code>[\$SERVER_NAME]</code> " on page 771
<code>[\$ShowBubble(width)]</code>	This is similar to <code>[\$ShowBubble]</code> . It calls the <code><ekbubbleinfo></code> tags and allows you to set the width of the bubble.	" <code>[\$ShowBubble(width)]</code> " on page 773
<code>[\$ShowBubble]</code>	Calls the <code><ekbubbleinfo></code> tags and places the information contained within those tags in a pop-up bubble.	" <code>[\$ShowBubble]</code> " on page 772
<code>[\$ShowContent('htmltagid')]</code>	Calls the <code><ekcontentinfo></code> tags and places the information in those tags within the specified HTML tag ID. Replace the 'htmltagid' with the ID of the tag.	" <code>[\$ShowContent('htmltagid')]</code> " on page 773
<code>[\$Status]</code>	Displays the status of a content item.	" <code>[\$Status]</code> " on page 774
<code>[\$Teaser]</code>	Display the content item's summary information.	" <code>[\$Teaser]</code> " on page 774
<code>[\$Title]</code>	Displays the content item's title.	" <code>[\$Title]</code> " on page 775
<code>[\$UrlEncode('str')]</code>	Encodes the string information. This variable can be used to encode another EkML variable and place it in an email.	" <code>[\$UrlEncode('str')]</code> " on page 777
<code>[\$UrlParam('paramname')]</code>	Displays the value of a querystring's parameter. For example, if the querystring is <code>?id=27</code> and the variable is <code>[\$UrlParam('id')]</code> , 27 is displayed.	" <code>[\$UrlParam('paramname')]</code> " on page 776

listsummary.ekml

This file defines which items and information are included when displaying content item information in a list summary using the ListSummary server control. For an example of using this file with a ListSummary server control, see the CMS400 demo example: `http://<your site>/CMS400Developer/Developer/ListSummary/TemplateMarkup.aspx`.

ListSummary.ekml Variables

Below is a list of variables that are used with the listsummary.ekml file.

Variable	Description	More Information
[\$Comment]	The content's comment information is displayed.	"[\$Comment]" on page 763
[\$ContentId]	Displays the content item's ID.	"[\$ContentId]" on page 763
[\$DateCreated]	Display the date and time the content was created.	"[\$DateCreated]" on page 764
[\$DateModified]	Display the date and time the content was last modified.	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	"[\$EditorLastName]" on page 766
[\$FolderDescription]	Displays the folder's description.	"[\$FolderDescription]" on page 786
[\$FolderId]	Display the folder ID of a content item.	"[\$FolderId]" on page 766
[\$FolderName]	Displays the folder's name.	"[\$FolderName]" on page 787
[\$Html]	Display the HTML contained in the content item.	"[\$Html]" on page 766
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	"[\$HyperLink]" on page 767
[\$Image]	Displays the path for the image defined in a content item's Metadata. When wrapped in <code></code> tag, the image is displayed.	"[\$Image]" on page 768
[\$ImageIcon]	Displays an image icon for the content item. For example, if the content item is HTML, the (📄) icon is displayed.	"[\$ImageIcon]" on page 768

Variable	Description	More Information
<code>[\$ImageThumbnail]</code>	Displays the path for the image's thumbnail defined in a content item's Metadata.	" <code>[\$ImageThumbnail]</code> " on page 769
<code>[\$Index]</code>	Serialize the content items in a numbered list.	" <code>[\$Index]</code> " on page 769
<code>[\$ItemCount]</code>	The total number of items in a list.	" <code>[\$ItemCount]</code> " on page 769
<code>[\$Language]</code>	Display the language ID for the content item.	" <code>[\$Language]</code> " on page 769
<code>[\$LinkTarget]</code>	When added to an <code></code> tag's <code>target=""</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	" <code>[\$LinkTarget]</code> " on page 770
<code>[\$QuickLink]</code>	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	" <code>[\$QuickLink]</code> " on page 770
<code>[\$SERVER_NAME]</code>	Displays the server name. For example, If this variable is applied to <code>http://www.example.com/demo.aspx</code> , the return is <code>www.example.com</code> .	" <code>[\$SERVER_NAME]</code> " on page 771
<code>[\$ShowBubble(width)]</code>	This is similar to <code>[\$ShowBubble]</code> . It calls the <code><ekbubbleinfo></code> tags and allows you to set the width of the bubble.	" <code>[\$ShowBubble(width)]</code> " on page 773
<code>[\$ShowBubble]</code>	Calls the <code><ekbubbleinfo></code> tags and places the information contained within those tags in a pop-up bubble.	" <code>[\$ShowBubble]</code> " on page 772
<code>[\$ShowContent('htmltagid')]</code>	Calls the <code><ekcontentinfo></code> tags and places the information in those tags within the specified HTML tag ID. Replace the 'htmltagid' with the ID of the tag.	" <code>[\$ShowContent('htmltagid')]</code> " on page 773
<code>[\$Status]</code>	Displays the status of a content item.	" <code>[\$Status]</code> " on page 774
<code>[\$Teaser]</code>	Display the content item's summary information.	" <code>[\$Teaser]</code> " on page 774
<code>[\$Title]</code>	Displays the content item's title.	" <code>[\$Title]</code> " on page 775

Variable	Description	More Information
<code>[\${UrlEncode('str')}]</code>	Encodes the string information. This variable can be used to encode another EkML variable and place it in an email.	" <code>[\${UrlEncode('str')}]</code> " on page 777
<code>[\${UrlParam('paramname')}]</code>	Displays the value of a querystring's parameter. For example, if the querystring is <code>?id=27</code> and the variable is <code>[\${UrlParam('id')}]</code> , 27 is displayed.	" <code>[\${UrlParam('paramname')}]</code> " on page 776

`[${FolderDescription}]`

Displays the folder's description.

example listsummary

This folder contains example content used with the List Summary feature.

1. Ektron Announces Winner of All-Stars Customer Competition
2. Ektron Expands Presence with Marketing, Interactive and Web Design Firms
3. Ektron Introduces an Enhanced Workflow Suite
4. Ektron Offers a Visual Development Environment for Rapid CMS Integration
5. Ektron Supports Rapid and Efficient Globalization Strategies on the Web
6. Ektron, Inc. Named One of New England's Fastest Growing Technology Companies
7. eWebEditPro+XML V4.2 supports the vision of create content once, reuse it many times

```
<ekmarkup>
  <ekoutput>
    <h3><b>[${FolderName}]</b></h3><p>[${FolderDescription}]<br/>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [${Index}]. [${HyperLink}]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```


[\$FolderName]

Displays the folder's name.

example_listsummary

This folder contains example content used with the List Summary feature.

1. Ektron Announces Winner of All-Stars Customer Competition
2. Ektron Expands Presence with Marketing, Interactive and Web Design Firms
3. Ektron Introduces an Enhanced Workflow Suite
4. Ektron Offers a Visual Development Environment for Rapid CMS Integration
5. Ektron Supports Rapid and Efficient Globalization Strategies on the Web
6. Ektron, Inc. Named One of New England's Fastest Growing Technology Companies
7. eWebEditPro+XML V4.2 supports the vision of create content once, reuse it many times

```
<ekmarkup>
  <ekoutput>
    <h3><b>[$FolderName]</b></h3><p/>[$FolderDescription]<br/>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            [$Index]. [$HyperLink]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

map.ekml

The map.ekml file defines which items and information are included when using the Map server control. Unlike most other .ekml files, the map.ekml has some variables that cannot be changed or moved around.

The file's variables appear in three <tr></tr> table rows. These rows are located below the main table. In the first two table rows, you can modify a tag's style information only. In the third, you can change the style information and the order of the variables to create different layouts for your page. It is recommended you

hide the first two table rows if you are not changing their style information. For example:

```
8 <ekmarkup>
9   <ekoutput>
10     <div>
11       <table>
12         <tr>...
58       <tr>...
87       <tr>
88         <td>
89           <table>
90             <tr>
91               <td>
92                 <div id="[$EkRouteInfoPane]" style="disp:
93               </td>
94             <td>
95               <div id="[$EkMap]" style="position:relat:
96             </td>
97             <td>
98               <div id="[$EkSearchTxtResultPane]" style=
99             </td>
100           </tr>
101         </table>
102       </td>
103     </tr >
104   </table>
105 </div>
106 </ekoutput>
107 </ekmarkup>
```

When using the map.ekml file remember these three rules.

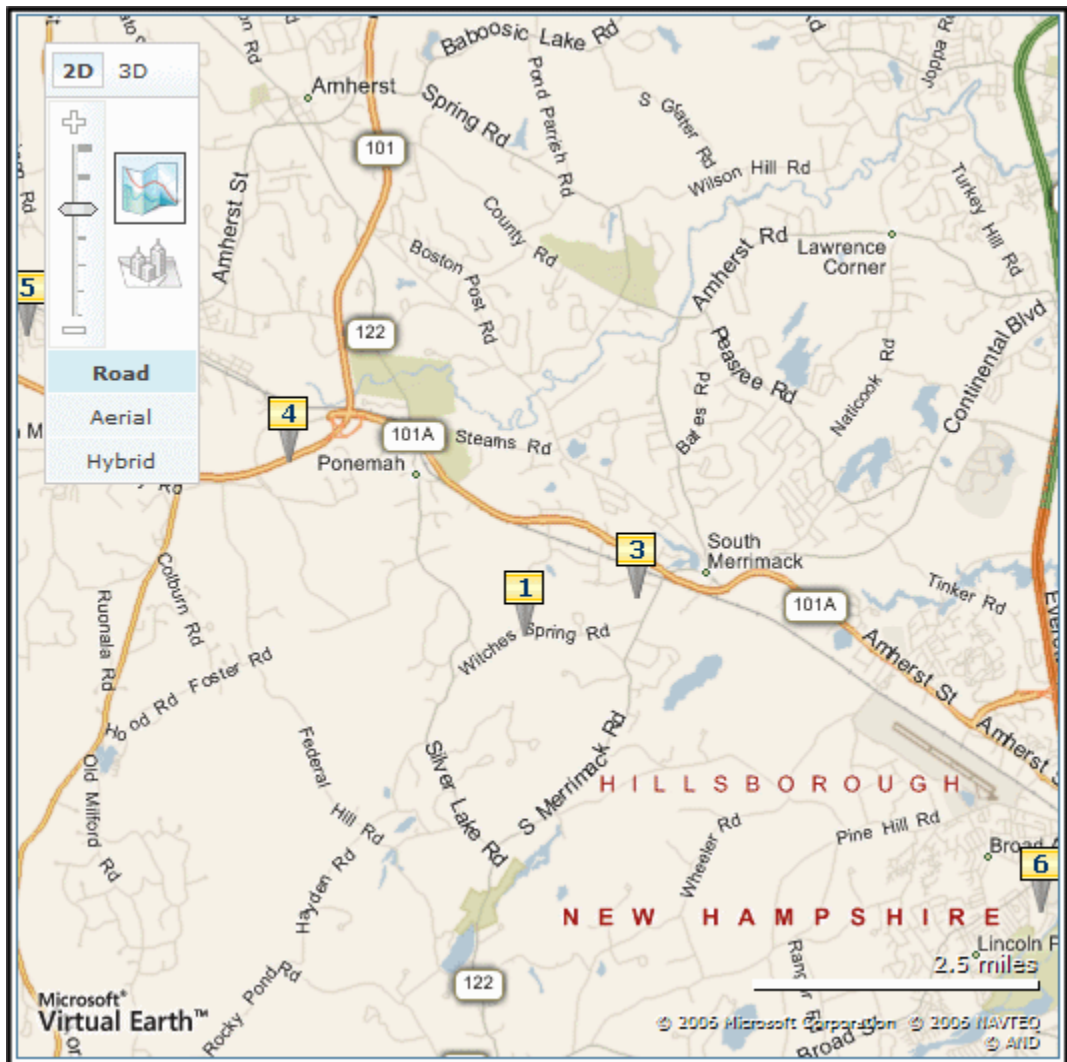
- 1. Do not change any of the IDs.
- 2. You can change any tag's style information.
- 3. You can move the variables in the third table row around to create different layouts. For example, you can display these sections horizontally or vertically.

Variable	Description	More Information
__Map	Displays the map section of the Map server control.	"__Map" on page 789

Variable	Description	More Information
__SearchTxtResultPane	Displays the results pane from the search.	"__SearchTxtResultPane" on page 790
__RouteInfoPane	Displays the direction information pane.	"__RouteInfoPane" on page 791













__Map

Displays the map section of the Map server control.



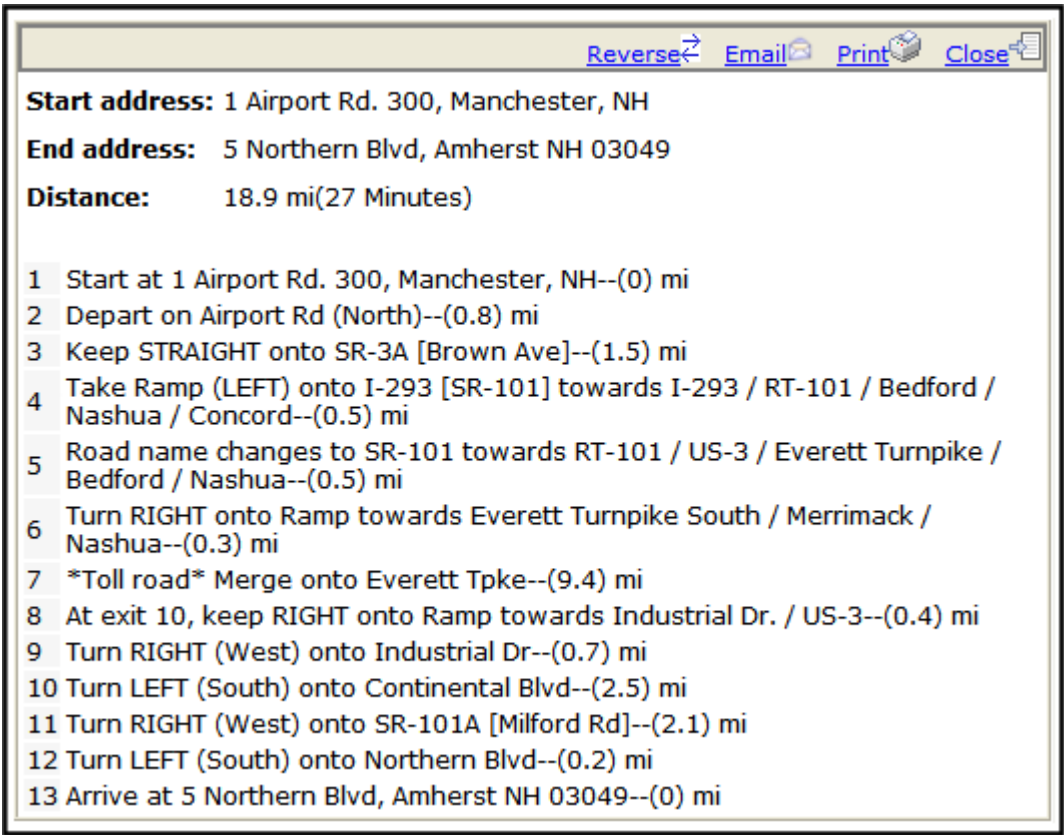
__SearchTxtResultPane

Displays the results pane from the search.

Results 1 - 6 of 6			
No.	Title	Distance	Map Direction
1.	Ektron Corporation 5 Northern Blvd, Amherst NH 03049 Ektron, Content Management Software	0.82	 
2.	Amherst House of Pizza 131 State Route 101A # 6, Amherst, NH 603) 886-5543	1.45	 
3.	Big a Pizza LLC 131 State Route 101A # 6, Amherst, NH	1.45	 
4.	Domino's Pizza 556 Nashua St, Milford, NH (603) 673-2700	3.83	 
5.	Pizza Top 183 Elm St, Milford, NH (603) 673-0037	7.89	 
6.	You You Japanese Bistro 150 Broad St #4, Nashua, NH 03063 Japanese, Korean, EurAsian Bistro	8.5	 

__RouteInfoPane

The panel that displays driving directions from the starting address to the arrival address.



metadatalist.ekml

This file defines which items and information are included when displaying content item information for a metadata list using the MetadataList server control.

metadatlist.ekml Variables

Below is a list of variables that are used with the metadatalist.ekml file.

Variable	Description	More Information
[\$Comment]	The content's comment information is displayed.	"[\$Comment]" on page 763

Variable	Description	More Information
[\$ContentId]	Displays the content item's ID.	"[\$ContentId]" on page 763
[\$DateCreated]	Display the date and time the content was created.	"[\$DateCreated]" on page 764
[\$DateModified]	Display the date and time the content was last modified.	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	"[\$EditorLastName]" on page 766
[\$FolderId]	Display the folder ID of a content item.	"[\$FolderId]" on page 766
[\$Html]	Display the HTML contained in the content item.	"[\$Html]" on page 766
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	"[\$HyperLink]" on page 767
[\$Image]	Displays the path for the image defined in a content item's Metadata. When wrapped in <code></code> tag, the image is displayed.	"[\$Image]" on page 768
[\$ImageIcon]	Displays an image icon for the content item. For example, if the content item is HTML, the () icon is displayed.	"[\$ImageIcon]" on page 768
[\$ImageThumbnail]	Displays the path for the image's thumbnail defined in a content item's Metadata.	"[\$ImageThumbnail]" on page 769
[\$Index]	Serialize the content items in a numbered list.	"[\$Index]" on page 769
[\$ItemCount]	The total number of items in a list.	"[\$ItemCount]" on page 769
[\$Language]	Display the language ID for the content item.	"[\$Language]" on page 769

Variable	Description	More Information
<code>[\$LinkTarget]</code>	When added to an <code></code> tag's <code>target=""</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	" <code>[\$LinkTarget]</code> " on page 770
<code>[\$QuickLink]</code>	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	" <code>[\$QuickLink]</code> " on page 770
<code>[\$SERVER_NAME]</code>	Displays the server name. For example, If this variable is applied to <code>http://www.example.com/demo.aspx</code> , the return is <code>www.example.com</code> .	" <code>[\$SERVER_NAME]</code> " on page 771
<code>[\$ShowBubble(width)]</code>	This is similar to <code>[\$ShowBubble]</code> . It calls the <code><ekbubbleinfo></code> tags and allows you to set the width of the bubble.	" <code>[\$ShowBubble(width)]</code> " on page 773
<code>[\$ShowBubble]</code>	Calls the <code><ekbubbleinfo></code> tags and places the information contained within those tags in a pop-up bubble.	" <code>[\$ShowBubble]</code> " on page 772
<code>[\$ShowContent('htmltagid')]</code>	Calls the <code><ekcontentinfo></code> tags and places the information in those tags within the specified HTML tag ID. Replace the 'htmltagid' with the ID of the tag.	" <code>[\$ShowContent('htmltagid')]</code> " on page 773
<code>[\$Status]</code>	Displays the status of a content item.	" <code>[\$Status]</code> " on page 774
<code>[\$Teaser]</code>	Display the content item's summary information.	" <code>[\$Teaser]</code> " on page 774
<code>[\$Title]</code>	Displays the content item's title.	" <code>[\$Title]</code> " on page 775
<code>[\$UrlEncode('str')]</code>	Encodes the string information. This variable can be used to encode another EkML variable and place it in an email.	" <code>[\$UrlEncode('str')]</code> " on page 777
<code>[\$UrlParam('paramname')]</code>	Displays the value of a querystring's parameter. For example, if the querystring is <code>?id=27</code> and the variable is <code>[\$UrlParam('id')]</code> , 27 is displayed.	" <code>[\$UrlParam('paramname')]</code> " on page 776

taxonomy.ekml

This file defines which items and information are included when displaying taxonomy item information using the Directory server control. Similar to maps.ekml, the taxonomy.ekml works differently than other .ekml files.

The Directory server control produces multiple areas where content or functionality is defined. To specify these areas in a template, you would call the `<ekoutput>` tag with the mode attribute equaling the area you want to define. For example, in the Directory server control, you can define information in the breadcrumb area of the server control by using `<ekoutput mode="breadcrumb">`.

Additional descriptions and commenting in the taxonomy.ekml file will assist you with learning about EkML with taxonomy. This file is located in `<web root>/CMS400Min/Workarea/template/taxonomy/taxonomy.ekml`.

Taxonomy `<ekoutput>` modes

The following `<ekoutput modes="">` are used with the taxonomy.ekml template.


EkML Tag	Description
<code><ekoutput mode="breadcrumb"></code>	<p>Defines the display of the breadcrumb portion of the taxonomy. Within these tags you can define:</p> <ul style="list-style-type: none"> • <code><bctitle></code> - title for the breadcrumbs • <code><bcrootlink></code> - root link text • <code><bcseparator></code> - separator used to between breadcrumbs in the breadcrumb trail • <code><bchyperlink></code> - hyperlinks in the breadcrumb trail • <code><bcactivelink></code> - the current active breadcrumb item
<code><ekoutput mode="category"></code>	<p>Defines how the category information is displayed. Within these tags you can define:</p> <p><code><ekcolrepeat></code> takes the category links and spreads them over the amount of columns that are defined in the Directory server control's TaxonomyCol property. This tag must appear within <code><ekrepeat></code> tags.</p>
<code><ekoutput mode="article_search"></code>	<p>Defines how the search results from the taxonomy are displayed.</p>

EkML Tag	Description
<code><ekoutput mode="article"></code>	Defines the display of an individual content item in your taxonomy.
<code><ekoutput mode="view"></code>	This section defines the overall output view of the Taxonomy. It denotes the locations of the Search box, Breadcrumb, Category and Articles.

taxonomy.ekml Variables

Below is a list of variables that are used with the taxonomy.ekml file.

Variable	Description	More Information
<code>[\$Category]</code>	The taxonomy category assigned to the content.	
<code>[\$Comment]</code>	The content's comment information is displayed.	"\$[\$Comment]" on page 763
<code>[\$ContentByteSize]</code>	Displays the content item's size in KB. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"\$[\$ContentByteSize]" on page 801
<code>[\$ContentId]</code>	Displays the content item's ID.	"\$[\$ContentId]" on page 763
<code>[\$DateCreated]</code>	Display the date and time the content was created.	"\$[\$DateCreated]" on page 764
<code>[\$DateModified]</code>	Display the date and time the content was last modified.	"\$[\$DateModified]" on page 765
<code>[\$EditorFirstName]</code>	Display the last editor's first name for a content item.	"\$[\$EditorFirstName]" on page 765
<code>[\$EditorLastName]</code>	Display the last editor's last name for a content item.	"\$[\$EditorLastName]" on page 766
<code>[\$FolderId]</code>	Display the folder ID of a content item.	"\$[\$FolderId]" on page 766
<code>[\$Html]</code>	Display the HTML contained in the content item.	"\$[\$Html]" on page 766

Variable	Description	More Information
[\$HyperLink]	Adds a hyperlink using the title of the content block as the text.	"[\$HyperLink]" on page 767
[\$Image]	Displays the path for the image defined in a content item's Metadata. When wrapped in <code></code> tag, the image is displayed.	"[\$Image]" on page 768
[\$ImageIcon]	Displays an image icon for the content item type. For example, if the content item is HTML, the () icon is displayed.	"[\$ImageIcon]" on page 768
[\$ImageThumbnail]	Displays the path for the image's thumbnail defined in a content item's Metadata.	"[\$ImageThumbnail]" on page 769
[\$Index]	Serialize the content items in a numbered list.	"[\$Index]" on page 769
[\$ItemCount]	The total number of items in a list.	"[\$ItemCount]" on page 769
[\$Language]	Display the language ID for the content item.	"[\$Language]" on page 769
[\$LinkTarget]	When added to an <code></code> tag's <code>target=" "</code> attribute, this variable reads the server control's <code>LinkTarget</code> property and uses its setting.	"[\$LinkTarget]" on page 770
[\$PagingCurrentEndIndex]	The end count number of the items on the page. For example, if you are displaying items 11 - 20 on a page, this variable represents the number 20. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"[\$PagingCurrentEndIndex]" on page 801
[\$PagingCurrentStartIndex]	The numerical record of the first item on a page. For example, if you are displaying items 1 through 10 out of 50 total items on a page, this variable represents the number 1. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"[\$PagingCurrentStartIndex]" on page 803
[\$QuickLink]	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	"[\$QuickLink]" on page 770

Variable	Description	More Information
[\$SearchDuration]	Displays the amount of time, in seconds, it has taken to execute the search. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"[\$SearchDuration]" on page 804
[\$SearchSummary]	Creates a summary from information stored in the indexing service for each item in the search results. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"[\$SearchSummary]" on page 800
[\$SearchText]	Displays the text for which a user is searching. This information is same as what a user entered in the search text box. Works only with taxonomy search. <code><ekoutput mode="article_search"></code>	"[\$SearchText]" on page 803
[\$SERVER_NAME]	Displays the server name. For example, If this variable is applied to <code>http://www.example.com/demo.aspx</code> , the return is <code>www.example.com</code> .	"[\$SERVER_NAME]" on page 771
[\$ShowAllcategory]	Adds to the Taxonomy search screen a checkbox that lets the user decide if he wants to display categories that have no items.	"[\$ShowAllcategory]" on page 771
[\$ShowBubble(width)]	This is similar to [\$ShowBubble]. It calls the <code><ekbubbleinfo></code> tags and allows you to set the width of the bubble.	"[\$ShowBubble(width)]" on page 773
[\$ShowBubble]	Calls the <code><ekbubbleinfo></code> tags and places the information contained within those tags in a pop-up bubble.	"[\$ShowBubble]" on page 772
[\$ShowContent('htmltagid')]	Calls the <code><ekcontentinfo></code> tags and places the information in those tags within the specified HTML tag ID. Replace the 'htmltagid' with the ID of the tag.	"[\$ShowContent('htmltagid')]" on page 773
[\$Status]	Displays the status of a content item.	"[\$Status]" on page 774
[\$Teaser]	Display the content item's summary information.	"[\$Teaser]" on page 774

Variable	Description	More Information
<code>[\$TemplateQuickLink]</code>	This property displays the Template Quicklink information assigned to the taxonomy item in the Workarea. When wrapped in an <code></code> tag, you can create a Hyperlink. <i>See Also:</i> Ektron CMS400.NET Administrator Manual section "Taxonomy" > "Creating a Taxonomy in the Workarea" > "Applying a Template to a Taxonomy or Category"	
<code>[\$Title]</code>	Displays the content item's title.	"\$Title" on page 775
<code>[\$UrlEncode('str')]</code>	Encodes the string information. This variable can be used to encode another EkML variable and place it in an email.	"\$UrlEncode('str')]" on page 777
<code>[\$UrlParam('paramname')]</code>	Displays the value of a querystring's parameter. For example, if the querystring is <code>?id=27</code> and the variable is <code>[\$UrlParam('id')]</code> , 27 is displayed.	"\$UrlParam('paramname')]" on page 776

[\$TemplateQuickLink]

This property displays the Template Quicklink information assigned to the taxonomy item in the Workarea. When wrapped in an `` tag, you can create a Hyperlink.

```
<ekmarkup>
  <ekoutput>
    <table width="100%" border="0">
      <ekrepeat>
        <tr>
          <td>
            <a href="[$TemplateQuickLink]">[Title]</a><br/>[$Teaser]
          </td>
        </tr>
      </ekrepeat>
    </table>
  </ekoutput>
</ekmarkup>
```

websearch.ekml


This file defines which items and information are included when displaying search results with the WebSearch server control.

When using websearch.ekml, you need two `<ekoutput>` nodes in the template. Each `<ekoutput>` node contains a result format that corresponds to a particular

type of search result. The first <ekoutput> node is the format used to display results for all non-images searches. The second <ekoutput> node determines the format of the results when searching for images.

websearch.ekml Variables

Below is a list of variables that are used with the websearch.ekml file.

Variable	Description	More Information
[\$ContentByteSize]	Displays the size of the content item.	"[\$ContentByteSize]" on page 801
[\$ContentId]	Display the content ID number assigned to the content.	"[\$ContentId]" on page 763
[\$DateModified]	Display the date and time the content was last modified.	"[\$DateModified]" on page 765
[\$EditorFirstName]	Display the last editor's first name for a content item.	"[\$EditorFirstName]" on page 765
[\$EditorLastName]	Display the last editor's last name for a content item.	"[\$EditorLastName]" on page 766
[\$Image]	Displays the path for the image defined in a content item's Metadata. When wrapped in tag, the image is displayed.	"[\$Image]" on page 768
[\$ImageIcon]	Used in non-image searches to display an image icon for the content item. For example, if the content item is HTML, the () icon is displayed.	"[\$ImageIcon]" on page 768
[\$ImageThumbnail]	Displays the path for the image's thumbnail defined in a content item's Metadata.	"[\$ImageThumbnail]" on page 769
[\$ItemCount]	Displays the total number of results produced by the search.	"[\$ItemCount]" on page 802
[\$LinkTarget]	When added to an tag's target=" " attribute, this variable reads the server control's LinkTarget property and uses its setting.	"[\$LinkTarget]" on page 770

Variable	Description	More Information
[\$PagingCurrentEndIndex]	The end count number of the items on the page. For example, if you are displaying items 11 - 20 on a page, this variable represents the number 20.	"[\$PagingCurrentEndIndex]" on page 801
[\$PagingCurrentStartIndex]	The numerical record of the first item on a page. For example, if you are displaying items 1 through 10 out of 50 total items on a page, this variable represents the number 1.	"[\$PagingCurrentStartIndex]" on page 803
[\$QuickLink]	This property displays the Quicklink information for the content item. When wrapped in an <code></code> tag, you can create a Hyperlink.	"[\$QuickLink]" on page 770
[\$SearchDuration]	Displays the amount of time, in seconds, it has taken to execute the search.	"[\$SearchDuration]" on page 804
[\$SearchSummary]	Creates a summary from information stored in the indexing service for each item in the search results.	"[\$SearchSummary]" on page 800
[\$SearchText]	Displays the text for which a user is searching. This information is same as what a user entered in the search text box.	"[\$SearchText]" on page 803
[\$Title]	Displays the content item's title.	"[\$Title]" on page 775


[\$SearchSummary]

Takes the first 300 characters from the body content and creates a summary.


For content items that do not get indexed, such as images, the SearchSummary uses the title of the image, the summary defined in the Workarea and metadata information.

Web

Results from template 1 - 10 of ektron for 50 . (0.37 seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)

Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)


Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"

[\$ContentByteSize]


Display the size of the content item in the results list.

Web

Results from template 1 - 10 of ektron for 50 . (0.37 seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)

Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)

Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"


[\$PagingCurrentEndIndex]


The numerical record of the last item on a page. For example, if you are displaying items 1 through 10 out of 50 total items on a page, this variable represents the number 10.

This variable is typically used in the following context:

Results [`[$PagingCurrentStartIndex]` - [`[$PagingCurrentEndIndex]`] of
[`[$ItemCount]`] for item [`[$SearchText]`] ([`[$SearchDuration]`]).

Web
Results from template 1 - 10 of ektron for 50 . (0.37 seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)
Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)
Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"

[`[$ItemCount]`]


Displays the total number of results produced by the search.

This variable is typically used in the following context:

Results [`[$PagingCurrentStartIndex]` - [`[$PagingCurrentEndIndex]`] of
[`[$ItemCount]`] for item [`[$SearchText]`] ([`[$SearchDuration]`])

Web
Results from template 1 - 10 of ektron for 50 (0.37 seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)
Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)
Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"

[\$SearchText]

Displays the text for which a user is searching. This information is same as what a user entered in the search text box.

This variable is typically used in the following context:

Results [\$PagingCurrentStartIndex] - [\$PagingCurrentEndIndex] of
[\$ItemCount] for item [\$SearchText] ([\$SearchDuration]).

**[\$PagingCurrentStartIndex]**


The numerical record of the first item on a page. For example, if you are displaying items 1 through 10 out of 50 total items on a page, this variable represents the number 1.


This variable is typically used in the following context:

Results [\$PagingCurrentStartIndex] - [\$PagingCurrentEndIndex] of
[\$ItemCount] for item [\$SearchText] ([\$SearchDuration]).

Web

Results from template **1** 10 of ektron for 50 . (0.37 seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)
Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)
Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"

[*\$SearchDuration*]


Displays the amount of time, in seconds, it has taken to perform the search.


This variable is typically used in the following context:

Results [*\$PagingCurrentStartIndex*] - [*\$PagingCurrentEndIndex*] of [*\$ItemCount*] for item [*\$SearchText*] (**[*\$SearchDuration*]**).

Web

Results from template 1 - 10 of ektron for 50 . (**0.37** seconds)


[Ektron Supports Rapid and Efficient Globalization Strategies on the Web](#)
(3/7/2006 2:00:46 PM)
Ektron Supports Rapid and Efficient Globalization Strategies on the Web. Efficient Globalization Strategies on the Web Powerful new tools in Ektrons CMS300 and CMS400.NET enable content managers to better handle end-to-end site translation and localization processes- Amherst, NH, December 7, 2004example_listsummary. ...
ID="284" Size="9 KB" LastAuthor="Application Administrator"


[Business Practices\(8/8/2006 5:55:50 PM\)](#)
Business Practices. Business Practice Standards ektron Medical's Global Business Practice Standards manual is a practical guide to business practice issues ektron Medical faces as a dynamic, growing corporation. It helps explain the universal principles governing business, provides clarity ab...
ID="84" Size="7 KB" LastAuthor="Application Administrator"

Ektron's Developer SDK

The Developer SDK for Ektron CMS400.NET contains the following components to help you extend and customize your CMS400.NET site.

- **The Developer API which includes:**
 - **Server Control API:** An interface for calling the methods and properties of the Ektron CMS400.NET Server Controls. For additional information on the server controls, see ["Introduction to Ektron CMS400.NET Server Controls" on page 7](#) and the Developer API Documentation.
 - **Web Services API:** Exposes a method's functionality for use with SOAP over HTTP. For additional information on the Web Services, see ["Web Services" on page 856](#) and the Developer API Documentation.
 - **.NET Assembly API:** Similar to the Business API provided in previous version, the .NET Assembly API provides an interface for calling the methods and properties that are exposed in Ektron CMS400.NET. See *Also:* the Developer API Documentation.
 - **Plug-in Extension API:** The Plug-in Extension exposes event hooks in Ektron CMS400.NET. As a developer, you can utilize these event hooks in Visual Studio C# or VB.NET to create customized events for your site. See *Also:* ["Plug-in Extension" on page 813](#) and the Developer API Documentation.

- **The Developer API Documentation**

The API Documentation contains a detailed description of the functions included in each of the APIs.

To access the Developer's API documentation in Visual Studio 2005, click **Help >> Contents**. Next, choose Ektron CMS400.NET API Documentation from the list of contents. You can also filter the documentation so you see only Ektron's API documentation. Click **Ektron API Documentation** in the filter drop down box. You can also access the API documentation online by clicking [here](#).

- **Plug-in Extension Wizard**

The Plug-in Extension exposes event hooks in Ektron CMS400.NET. As a developer, you can utilize these event hooks in Visual Studio C# or VB.NET to create customized events for your site. The Plug-in Extension Wizard creates the framework code in Visual Studio 2005 to make an extension. All you need to do is add your custom code to make your events happen. See *Also:* ["Plug-in Extension" on page 813](#)

- **Ektron CMS400.NET Server Controls Toolbox**

The Ektron CMS400.NET's are now install for you when the Developer SDK is installed. Server controls let you insert, via drag and drop or programmatically, many standard methods and properties within the Visual Studio 2005 environment. This means that you can see the effect of your changes in real time -- you don't have to modify a page then compile a

sample project to see the results. See Also: ["Introduction to Ektron CMS400.NET Server Controls" on page 7](#)

- **Ektron Site Setup Wizard**

The Ektron Site Setup Wizard allows you to create a site in Visual Studio 2005. This allows you to easily create additional CMS400.NET sites on your server. The site wizard copies the site file needed for a working CMS400.NET Min site. In addition, it adds all the necessary permissions, sets up the database, creates the indexing catalog and creates the asset storage location. See Also: ["Using the Ektron Site Setup Wizard" on page 806](#)

Installing the Developer SDK

To install the Developer SDK on your Ektron CMS400.NET Server, run the CMS400SDK_Setup.exe file located in
C:\Program Files\Ektron\CMS400v7x\Utilities.

NOTE If you installed the Developer SDK on your server during the initial CMS400.NET install, you do not have to install it again.

If you are installing to a separate client system, copy the CMS400SDK_Setup.exe to that system and run it.

Use the steps below to guide you through the install procedure.

1. Run CMS400SDK_Setup.exe.
2. The software is installed and configured.
3. When the SDK setup is complete, click **Finish**.

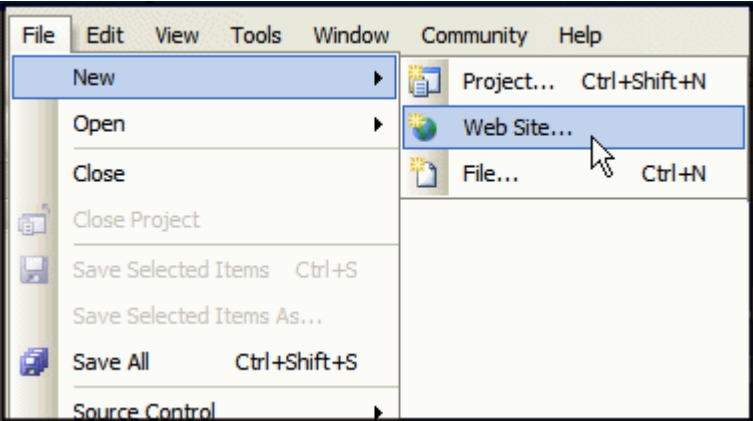
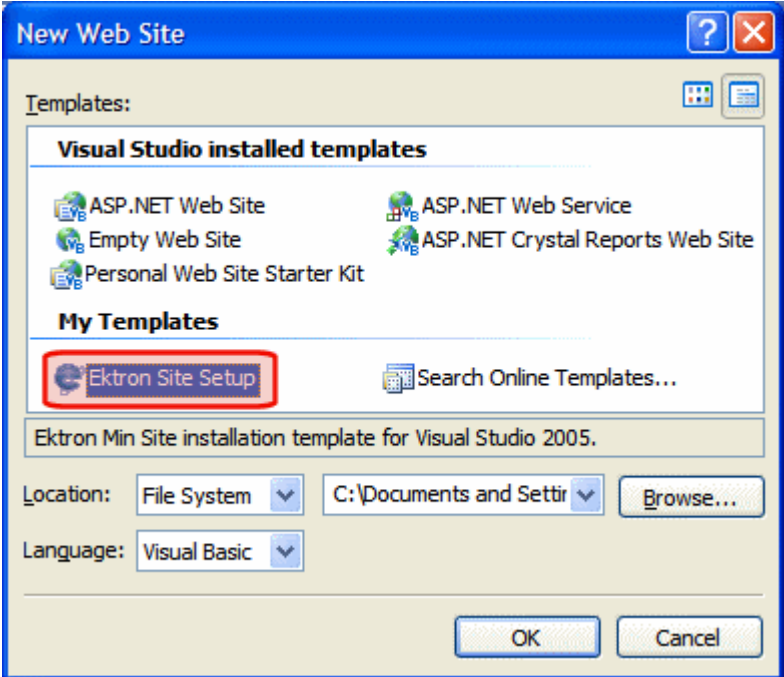
Using the Ektron Site Setup Wizard

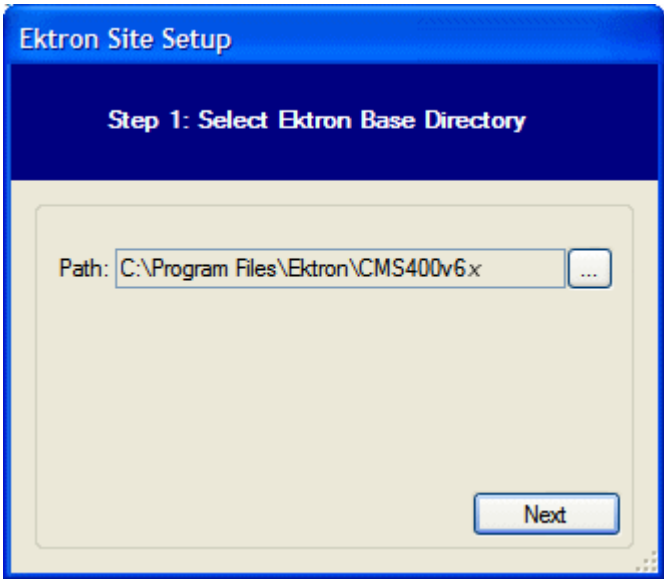
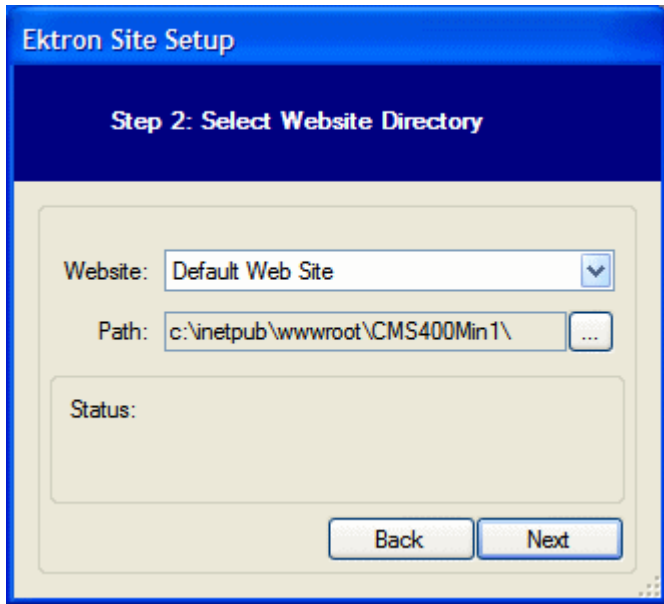
The Ektron Site Setup Wizard walks you through the steps to create a Min site in Visual Studio 2005. The purpose of a Min site is that you start with a basic site and customize it from there.

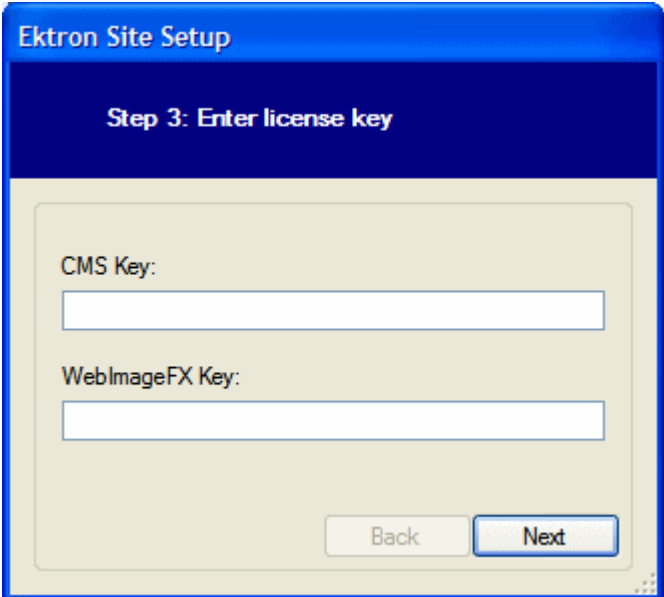
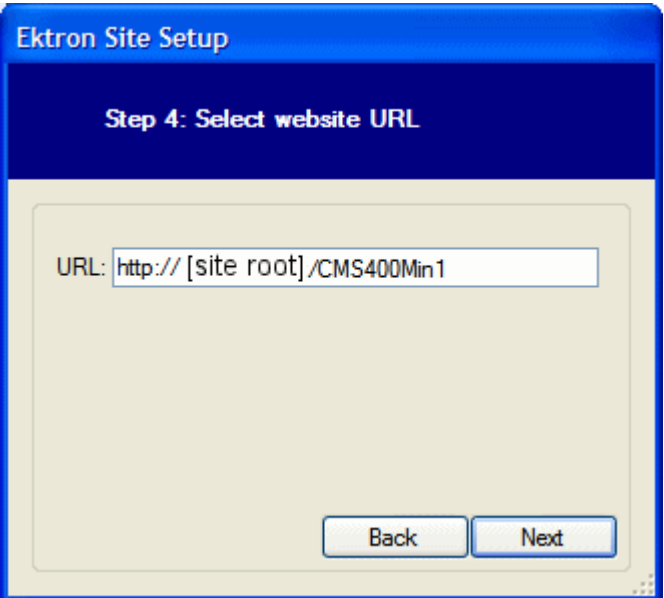
The Ektron Site Setup Wizard:

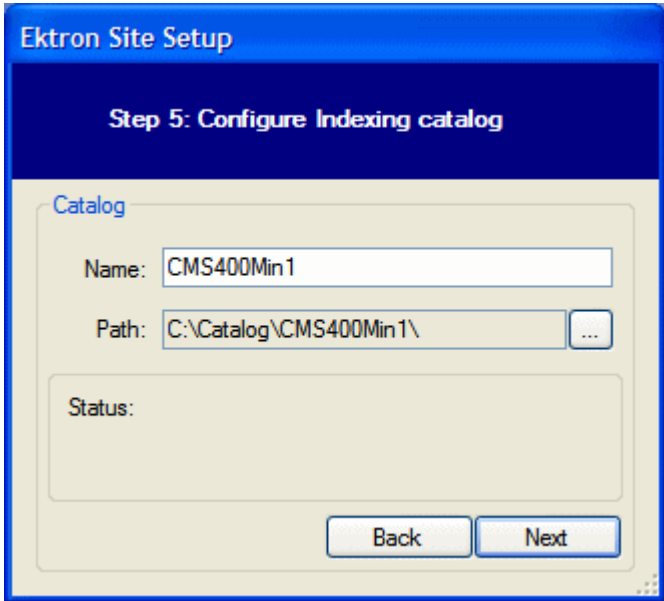
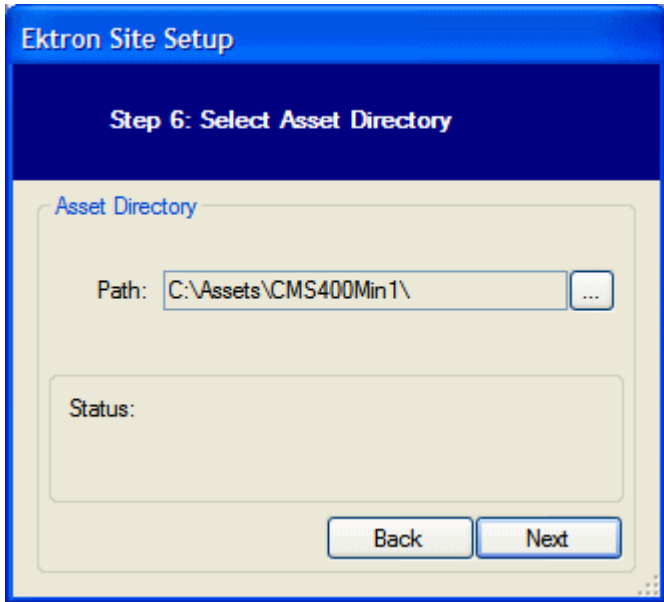
- creates a new Min site on the server
- sets the permissions
- setup the indexing catalog and service
- creates a database for the site
- creates a new Asset location for the site

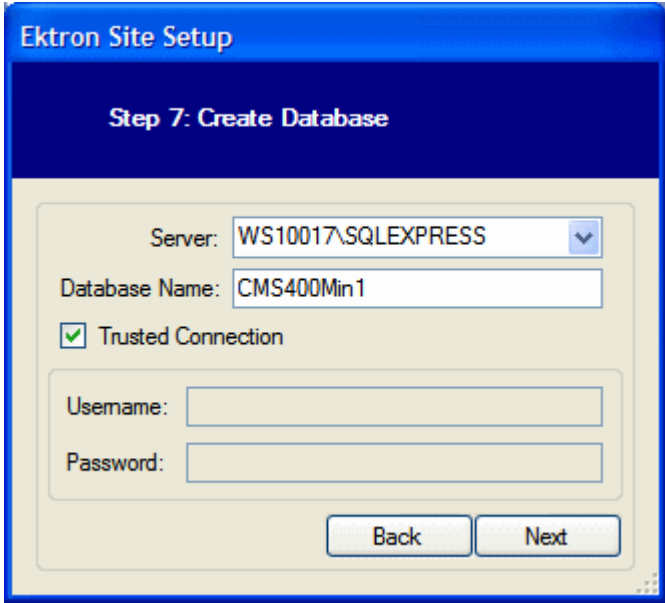
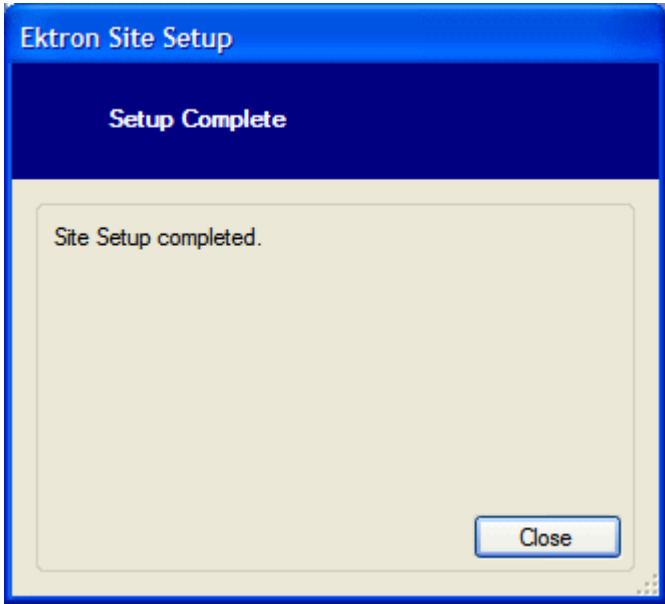
Steps to Use the Ektron Site Setup Wizard

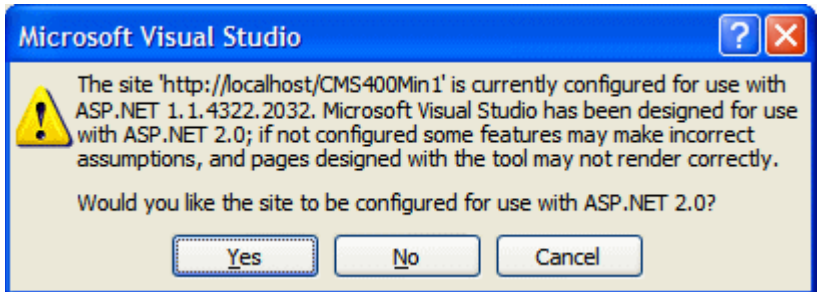
Step	Setup Screen
<p>1. In Visual Studio 2005, click File > New > Web Site.</p>	 <p>The screenshot shows the Visual Studio 2005 File menu. The 'New' option is expanded, and 'Web Site...' is highlighted. Other visible options include 'Project...', 'File...', and 'Source Control'.</p>
<p>2. In the New Web Site window:</p> <ul style="list-style-type: none"> • click Ektron Site Setup. • choose the Location • choose the Language • click Next 	 <p>The screenshot shows the 'New Web Site' dialog box. Under 'Visual Studio installed templates', 'ASP.NET Web Site' is selected. Under 'My Templates', 'Ektron Site Setup' is highlighted with a red rectangle. The 'Location' is set to 'File System' and 'C:\Documents and Settings\...', and the 'Language' is set to 'Visual Basic'. The 'OK' button is visible at the bottom right.</p>

Step	Setup Screen
<p>3. Select the Ektron CMS400.NET base directory and click Next.</p> <hr/> <p>Note: This is where Ektron CMS400.NET is installed.</p> <hr/>	 <p>The screenshot shows the 'Ektron Site Setup' window, Step 1: Select Ektron Base Directory. It features a text box for 'Path' containing 'C:\Program Files\Ektron\CMS400v6x' and a 'Next' button at the bottom right.</p>
<p>4. Select the new Web site's directory and click Next.</p> <p>The necessary files are copied to the appropriate location.</p>	 <p>The screenshot shows the 'Ektron Site Setup' window, Step 2: Select Website Directory. It includes a 'Website' dropdown menu set to 'Default Web Site', a 'Path' text box with 'c:\inetpub\wwwroot\CMS400Min1\' and a browse button, and a 'Status' label. 'Back' and 'Next' buttons are at the bottom.</p>

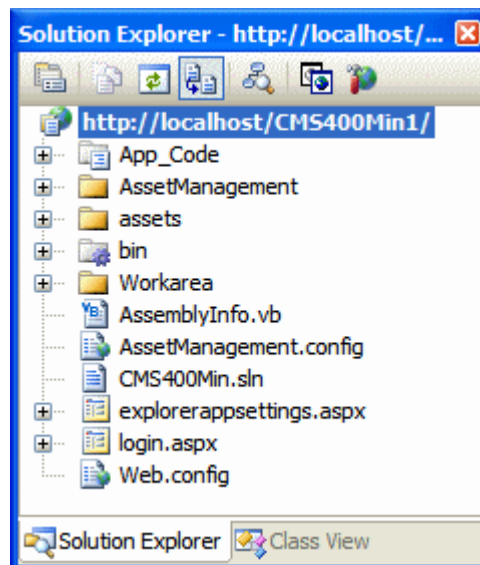
Step	Setup Screen
<p>5. Enter your CMS400.NET and WebImageFX license keys. Click Next.</p>	 <p>The screenshot shows a window titled "Ektron Site Setup" with a dark blue header. Below the header, it says "Step 3: Enter license key". The main area is light beige and contains two text input fields. The first is labeled "CMS Key:" and the second is labeled "WebImageFX Key:". At the bottom right, there are two buttons: "Back" and "Next".</p>
<p>6. Enter the URL for the site and click Next.</p>	 <p>The screenshot shows a window titled "Ektron Site Setup" with a dark blue header. Below the header, it says "Step 4: Select website URL". The main area is light beige and contains a single text input field labeled "URL:". The text inside the field is "http:// [site root] /CMS400Min1". At the bottom right, there are two buttons: "Back" and "Next".</p>

Step	Setup Screen
<p>7. Choose the Name and Path for the Indexing Catalog. Click Next.</p>	 <p>The screenshot shows the 'Ektron Site Setup' window at 'Step 5: Configure Indexing catalog'. The 'Catalog' section has a 'Name' field with 'CMS400Min1' and a 'Path' field with 'C:\Catalog\CMS400Min1\'. There is a 'Status' label below the path field. At the bottom are 'Back' and 'Next' buttons.</p>
<p>8. Select the Path for the Asset directory and click Next.</p>	 <p>The screenshot shows the 'Ektron Site Setup' window at 'Step 6: Select Asset Directory'. The 'Asset Directory' section has a 'Path' field with 'C:\Assets\CMS400Min1\'. There is a 'Status' label below the path field. At the bottom are 'Back' and 'Next' buttons.</p>

Step	Setup Screen
<p>9. Create a Database for the site:</p> <ul style="list-style-type: none">• select your DB server from the Server drop-down list• create a Database Name• if you are using a Trusted Connection, make sure a green check is in check box• If you are not using a Trusted Connection, uncheck the check box and add a valid User-name and Password• Click Next	 <p>The screenshot shows the 'Ektron Site Setup' window at 'Step 7: Create Database'. It features a 'Server' dropdown menu set to 'WS10017\SQLEXPRESS', a 'Database Name' text box containing 'CMS400Min1', and a checked 'Trusted Connection' checkbox. Below these are empty text boxes for 'Username' and 'Password'. At the bottom right are 'Back' and 'Next' buttons.</p>
<p>10. When the Setup Complete screen appears click Close</p>	 <p>The screenshot shows the 'Ektron Site Setup' window at the 'Setup Complete' screen. It displays the message 'Site Setup completed.' in a large text area. A 'Close' button is located at the bottom right of the window.</p>

Step	Setup Screen
11. In Visual Studio 2005 a pop-up appears asking if you would like to have the site configured to use ASP.NET 2.0. Click Yes .	

Once the Setup is complete, your new site appears in the Solution Explorer. From this point, you can create and add to your new site as you would and existing site.



Plug-in Extension

Introduction

The Plug-in Extension exposes event hooks in Ektron CMS400.NET. As a developer, you can utilize these event hooks in Visual Studio C# or VB.NET to create customized events for your site. For instance, you could create a Plug-in Extension that:

- sends an instant message when content is published
- sends an email when content is published
- filters out objectionable language from blog posts
- appends HTML content when it's published

How a Plug-in Works with CMS400.NET

The following example explains the way a plug-in works. This example uses the `OnBeforePublish` event to append HTML content with copyright information before it is published.

A content editor saves a piece of content in Ektron CMS400.NET. Before the content is published, the `OnBeforePublish` event is fired. The Extension Architecture checks a watch folder to see if there are any plug-ins that have the `OnBeforePublish` event. The Extension Architecture sees the `OnBeforePublish` event and executes the code. The content is then appended with the copyright information and published.

The Plug-in Extension Architecture and the Plug-in Wizard

The Plug-in Extension is comprised of two parts. The first, the Plug-in Extension Architecture, is installed on your server during the Ektron CMS400.NET install. You can control whether the Plug-in Extension Architecture is active or not by changing the `ek_extensionServiceEnabled` key in the `Web.config`. Setting this element to `True` activates the extension service. The key is set to `True` by default.

The second part, the Plug-in Extension Wizard, is installed when you run the Developer SDK install on your development system. See *Also: "Installing the Developer SDK" on page 814 and "Ektron's Developer SDK" on page 805.*

This wizard creates a Plug-in Extension. In that extension is the framework code for events. All you need to do is add your custom code to the event framework. Once the code is built into a DLL, you move the DLL to the Extensions watch folder.

Managing your Plug-ins

Ektron, Inc. also provides a Plug-in Extension configuration manager to help manage and prioritize which Plug-in Extensions are executed. This configuration utility also allows you to disable or enable plug-ins, decide what should happen in

the case of an unhandled error and choose which sites can use the plug-in. See [Also: "Configuring a Plug-in Extension" on page 823](#)

Creating a Plug-in Overview

The table below provides an overview of the steps involved when creating a plug-in.

Plug-in Extension Overview Steps	See Also
1. Decide what you would like the plug-in to accomplish. For example, you want to add copyright information to each piece of HTML content that is published.	
2. Create the plug-in extension using the Plug-in Extension Wizard in Visual Studio 2005 C# or Visual Basic.	"Creating a New Plug-in Extension in Visual Studio 2005" on page 815
3. Add your code to the Plug-in Extension's framework. The Plug-in Samples section of this chapter provides samples.	"Plug-in Samples" on page 819
4. Move the newly created Plug-in Extension DLL to the watch folder on your Ektron CMS400.NET server located at: C:\Program Files\Ektron\Plugins\Extensions	
5. Use the Ektron Extensibility Configuration Tool to further configure your Plug-in Extensions.	"Configuring a Plug-in Extension" on page 823

The following sections are contained in this chapter.

- ["Installing the Developer SDK" on page 814](#)
- ["Creating a New Plug-in Extension in Visual Studio 2005" on page 815](#)
- ["Plug-in Samples" on page 819](#)
- ["Manually Creating Plug-in Framework" on page 823](#)
- ["Configuring a Plug-in Extension" on page 823](#)
- ["Connecting to the Web Service API From a Plug-in" on page 836](#)
- ["Debugging a Plug-in" on page 837](#)

Installing the Developer SDK

To install the Developer SDK on your Ektron CMS400.NET Server, run the CMS400SDK_Setup.exe file located in
C:\Program Files\Ektron\CMS400v7x\Utilities.

NOTE If you installed the CMS400.NET SDK on your server during the initial install, you do not have to install it again.

If you are installing to a separate client system, copy the CMS400SDK_Setup.exe to that system and then run it.

Use the steps below to guide you through the install procedure.

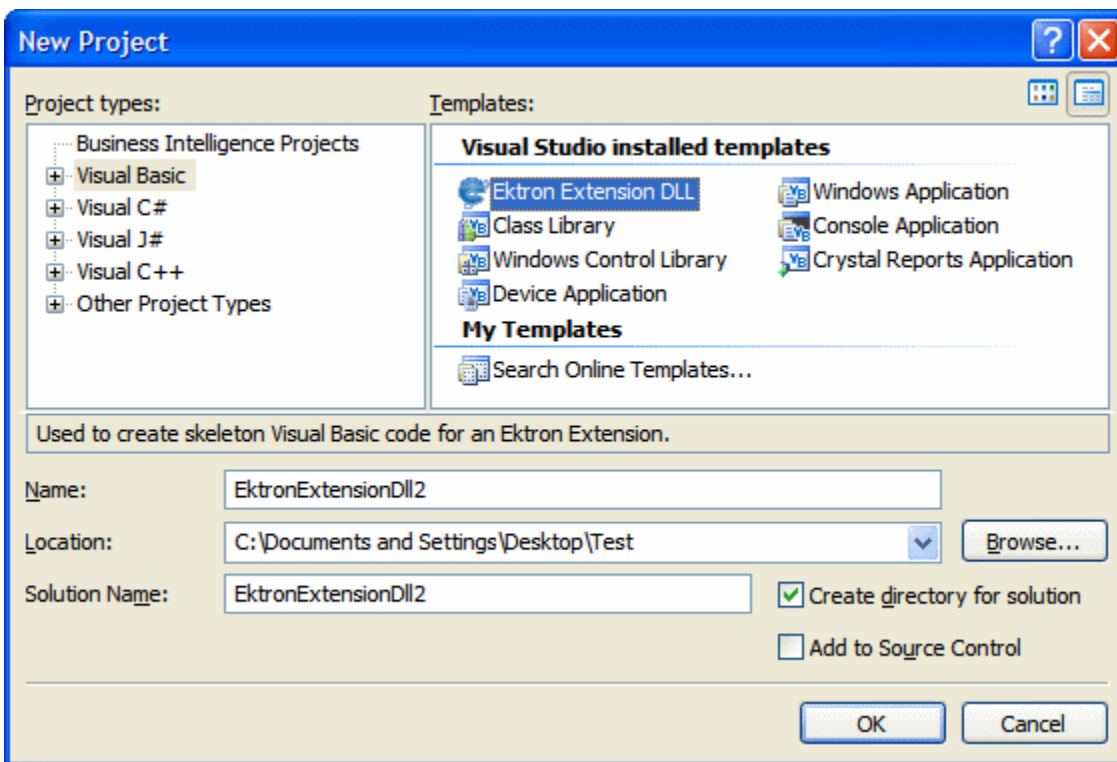
1. Run CMS400SDK_Setup.exe.
2. The software is installed and configured.
3. When the SDK setup is complete, click **Finish**.

Creating a New Plug-in Extension in Visual Studio 2005

WARNING! Do not Dotfuscate plug-in extensions. When CMS400.NET fires an event, the plug-in extension architecture needs to read the plug-in to see if there is a matching event. Dotfuscation encrypts the code in a plug-in and the plug-in extension architecture is not able to read it.

The following steps explain how to create a new Plug-in Extension in Visual Studio 2005.

1. In Visual Studio 2005, click **File > New > Project...**
2. The New Project screen appears.



3. Under **Project Type**, choose either **Visual Basic** or **Visual C#**.

4. Under **Templates**, click **Ektron Extension DLL**.
5. Fill out the **Name**, **Location** and **Solution Name** text boxes.
6. Click **OK**.
7. The Ektron Extension Template screen appears.

Ektron Extension Template

Events

Content Events

- ☐ OnBeforeContentAdd ☐ OnAfterContentAdd
- ☐ OnBeforeContentEdit ☐ OnAfterContentEdit
- ☐ OnBeforeContentDelete ☐ OnAfterContentDelete
- ☐ OnBeforePublish ☐ OnAfterPublish

Folder Events

- ☐ OnBeforeFolderAdd ☐ OnAfterFolderAdd
- ☐ OnBeforeFolderEdit ☐ OnAfterFolderEdit
- ☐ OnBeforeFolderDelete ☐ OnAfterFolderDelete

User Events

- ☐ OnBeforeUserAdd ☐ OnAfterUserAdd
- ☐ OnBeforeUserEdit ☐ OnAfterUserEdit
- ☐ OnBeforeUserDelete ☐ OnAfterUserDelete
- ☐ OnBeforeUserLogin ☐ OnAfterUserLogin
- ☐ OnBeforeFormSubmit ☐ OnAfterFormSubmit

Cancel Next >

8. Select the applicable Extension Methods. For descriptions of the methods, use the table below.

Event	This Hook is Called...	Returns
Content Events		

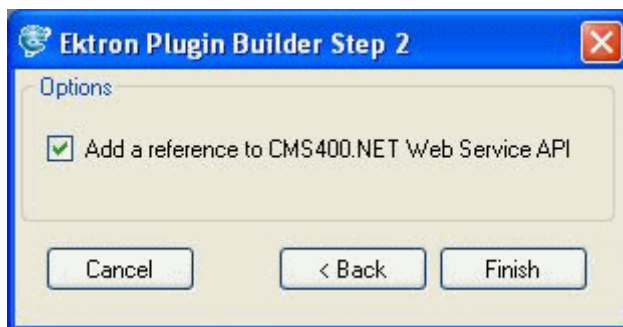
Event	This Hook is Called...	Returns
OnBeforeContentAdd	Before content is added.	A boolean indicating success or failure. False means failure.
OnBeforeContentEdit	Before content is edited.	A boolean indicating success or failure. False means failure.
OnBeforeContentDelete	Before content is deleted.	A boolean indicating success or failure. False means failure.
OnBeforePublish	Before content is published.	A boolean indicating success or failure. False means failure.
OnAfterContentAdd	After content is added.	A boolean indicating success or failure. False means failure.
OnAfterContentEdit	After an edit is completed.	A boolean indicating success or failure. False means failure.
OnAfterContentDelete	After content is deleted.	A boolean indicating success or failure. False means failure.
OnAfterPublish	After content is published.	A boolean indicating success or failure. False means failure.
Folder Events		
OnBeforeFolderAdd	Before a folder is created.	A boolean indicating success or failure. False means failure.
OnBeforeFolderEdit	Before a folder is edited.	A boolean indicating success or failure. False means failure.
OnBeforeFolderDelete	Before a folder is deleted.	A boolean indicating success or failure. False means failure.
OnAfterFolderAdd	After a folder is created.	A boolean indicating success or failure. False means failure.
OnAfterFolderEdit	After a folder is edited.	A boolean indicating success or failure. False means failure.
OnAfterFolderDelete	After a folder is deleted.	A boolean indicating success or failure. False means failure.

Event	This Hook is Called...	Returns
User Events		
OnBeforeUserAdd	Before a user is created.	A boolean indicating success or failure. False means failure.
OnBeforeUserEdit	Before a user is edited.	A boolean indicating success or failure. False means failure.
OnBeforeUserDelete	Before a user is deleted.	A boolean indicating success or failure. False means failure.
OnBeforeUserLogin	Before a user logs in to the CMS.	A boolean indicating success or failure. False means failure.
OnBeforeFormSubmit	Before form data is submitted to the CMS database.	A boolean indicating success or failure. False means failure.
OnAfterUserAdd	After a user is created.	A boolean indicating success or failure. False means failure.
OnAfterUserEdit	After a user is edited.	A boolean indicating success or failure. False means failure.
OnAfterUserDelete	After a user is deleted.	A boolean indicating success or failure. False means failure.
OnAfterUserLogin	After a user logs in to the CMS.	A boolean indicating success or failure. False means failure.
OnAfterFormSubmit	After form data is submitted to the CMS database.	A boolean indicating success or failure. False means failure.

NOTE [OnBeforeReviewRejected](#) and [OnAfterReviewRejected](#) are available when [Manually Creating Plug-in Framework](#). See ["Manually Creating Plug-in Framework"](#) on page 823.

9. Click **Next**.

10. In the Ektron Plug-in Builder Step 2 screen, place a check mark in the box to add a reference to the CMS400.NET Web Service API. Check this box if you are planning to use Web Service Proxies. See *Also*: "[Connecting to the Web Service API From a Plug-in](#)" on page 836



11. Click **Finish**.
12. A new project is created in Visual Studio 2005 with the beginning framework for each method you chose.
13. Add the appropriate code to the method's framework.

NOTE For samples of code, see "[Plug-in Samples](#)" on page 819.

14. Save and build the project.
15. Navigate to the location of your Plug-in Extension project.
16. From the Bin folder of the Plug-in Extension project, copy the newly created DLL to the following folder on your Ektron CMS400.NET server.

C:\Program Files\Ektron\Plugins\Extensions

Once you have created a plug-in, you can use the Extensibility Configuration Editor to configure how the plug-in will interact with CMS400.NET. See *Also*: "[Configuring a Plug-in Extension](#)" on page 823

Plug-in Samples

Ektron provides the following sample code plug-in projects. These projects are located in the C:\Program Files\Ektron\CMS400SDK\Samples folder. Use the `Samples.sln` file to launch these samples in Visual Studio 2005.

- PublishSendAlert
- PublishContentChange
- InstantMessageOnPublish

The **PublishSendAlert** sample sends an email when new content is published. For example, you want to be notified if any content is published on the Web site. See "[PublishSendAlert Code Sample](#)" on page 820

The **PublishContentChange** sample appends content when it is published. For example, you want to add a signature or copyright information to content when it is published. See ["PublishContentChange Sample" on page 821](#)

The **InstantMessageOnPublish** sample sends an instant message when new content is published. For example, you want to be notified if any content is published on the Web site. See ["InstantMessageOnPublish Sample" on page 822](#)

The code for each sample appears below.

PublishSendAlert Code Sample

The following code sample sends an email when new content is published.

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Net.Mail;
using System.Net;
using Ektron.Cms.Extensibility;

namespace Ektron.Cms.Extensibility.Samples
{
    public class PublishSendAlert : ExtensionEvent
    {
        public override bool OnAfterPublish()
        {
            // Declare variables.
            bool bReturn = false;
            SmtpClient sendClient = null;
            MailMessage message = null;
            MailAddress from = null;

            try
            {
                // Create connection to mail server and create a new message.
                sendClient = new SmtpClient("mail.sample.com", 25);
                message = new MailMessage();

                // Populate information in the new message.
                // TODO: Add from address below
                from = new MailAddress("sample_sender@sample.com");

                message.From = from;

                // TODO: Add all recipients below.
                message.To.Add("email@sample.com");
                message.To.Add("email2@sample.com");
                message.Subject = "Content has been published";
                message.Body = "Content published contained the
following:\r\n" + Content.Html;

                // Send the message.
                sendClient.Send(message);

                // Set return to success.
                bReturn = true;
            }
        }
    }
}
```

```

        catch (Exception exThrown)
        {

            // If there was an error, catch it and record the last error
            // message. This is displayed in the CMS to the end user based upon the
            // method return value and the setup in the configuration utility.
            // Typically SmtpClient throws an exception if the message could not be
            // sent.
            Content.ErrorMessage = exThrown.Message + exThrown.StackTrace;
            bReturn = false;
        }
        finally
        {
            // Cleanup
            if (message != null)
            {
                message.Dispose();
                message = null;
            }
            if (sendClient != null)
            {
                sendClient = null;
            }
        }

        return bReturn;
    }
}

```

PublishContentChange Sample

The following code sample appends HTML content when it is published.

IMPORTANT!

Only HTML content can be appended using the OnBeforePublish event. Assets can not appended and there is no error message.

```

using System;
using System.Collections.Generic;
using System.Collections;
using System.Text;
using Ektron.Cms.Extensibility;

namespace Ektron.Cms.Extensibility.Samples
{
    public class PublishContentChange : ExtensionEvent
    {
        private const string S_APPEND = " ****added by
PublishContentChange****";

        public override bool OnBeforePublish()
        {
            // Declare variables
            bool bReturn = false;
            string sLastError = "";

```

```

        string sTemp = Content.Html;

        try
        {
            //Check content for the last closing paragraph tag. If one exists
            //put additional text within it. Otherwise just append to end of
            //content.
            if (sTemp.Contains("</p>"))
            {
                sTemp = sTemp.Insert(Content.Html.LastIndexOf("</p>"), S_APPEND);
            }
            else
            {
                sTemp += S_APPEND;
            }

            // Set content html to our new updated string. This will set it on
            // the CMS side.
            Content.Html = sTemp;

            // Set return to success.
            bReturn = true;
        }
        catch (Exception exThrown)
        {
            // If there was an error, catch it and record the last error
            // message. This is displayed in the CMS to the end user based upon the
            // method return value and the setup in the configuration utility.
            Content.ErrorMessage = exThrown.Message +
            exThrown.StackTrace;
            bReturn = false;
        }
        return bReturn;
    }
}

```

InstantMessageOnPublish Sample

This sample plug-in sends an instant message (IM) to any designated AIM user. This sample is made up of various files. To view these files, navigate to the C:\Program Files\Ektron\CMS400SDK\Samples\InstantMessageOnPublish folder.

Use the following steps when working with this sample.

1. Open the config_info.reg file located in the C:\Program Files\Ektron\CMS400SDK\Samples\InstantMessageOnPublish folder and edit the following lines:
 - change the SendToUserName
 - change the SendFromUserName
 - change the SendFromUserPassword
2. Save the file.
3. Double click the config_info.reg to add the information to the registry.
4. In Visual Studio 2005, open the Sample.sln located at: C:\Program Files\Ektron\CMS400SDK\Samples

5. Save and build the project.
6. Navigate to the location of your InstantMessageOnPublish plug-in project.
7. From the Bin folder of the plug-in project, copy the newly created DLL to the following folder on your Ektron CMS400.NET server.

C:\Program Files\Ektron\Plugins\Extensions

Once you have created a plug-in, you can use the Extensibility Configuration Editor to configure how the plug-in will interact with CMS400.NET. See Also: ["Configuring a Plug-in Extension" on page 823](#)

Manually Creating Plug-in Framework

If you would like to create a plug-in without using the Plug-in Extension Wizard or you cannot use the wizard, the following steps explain how to convert a standard DLL to an Ektron Plug-in DLL.

1. Create a DLL project skeleton via the VS Class Library wizard.
2. Add a reference to ExtensibilityBase.dll from your current project.
The default install location is: C:\Program Files\Ektron\CMS400SDK
3. Add a using statement for C# or an Imports statement for VB to incorporate the plug-in namespace.

C#: `using Ektron.Cms.Extensibility;`

VB: `Imports Ektron.Cms.Extensibility`

4. Add inheritance to your existing class from ExtensionEvent base class.

C#: `public class MyStandardDllClass : ExtensionEvent`

VB: `Inherits ExtensionEvent`

5. Hook the events you want to implement by using intellisense to choose the event. Type:

C#: `public override`

VB: `Public Overrides`

6. If the plug-in is to return a status other than True, remove the return line generated by visual studio and return your own status:

C#: `return base.<Overriden_Method_Name>();`

VB: `Return MyBase.<Overriden_Method_Name>()`

Once you have created a plug-in, you can use the Extensibility Configuration Editor to configure how the plug-in will interact with CMS400.NET. See Also: ["Configuring a Plug-in Extension" on page 823](#)

Configuring a Plug-in Extension

Once you have created a Plug-in Extension, the Extensibility Configuration Editor allows you to:

- prioritize plug-ins and events
- enable or disable them
- assign them to a specific site when using multiple sites with CMS400.NET

- see a description of the plug-in or event
- see who created a plug-in or event

By prioritizing the plug-ins and events you can set the order of execution for events of the same type that appear in multiple plug-ins. *See Also:* ["Setting the Plug-in Order of Execution by Event" on page 835](#)

Enabling and disabling plug-ins allows you to temporarily disable a plug-in or event without disabling the rest of the plug-ins and events. *See Also:* ["Enabling and Disabling Plug-ins and Events" on page 834](#)

If you are using the Multi-site support in CMS400.NET, you can assign a plug-in or event to a specific site. This allows you to build a Plug-in that might be used in multiple sites, but contain an event that you only want applied to a single site. *See Also:* ["Applying Plug-ins and Events to Specific Sites" on page 830](#)

The Extensibility Configuration Editor exists on your CMS400.NET server. For the editor to work with an Extension Plug-in, the plug-in needs to be in the watch folder `C:\Program Files\Ektron\Plugins\Extensions` located on the Ektron CMS400.NET server.

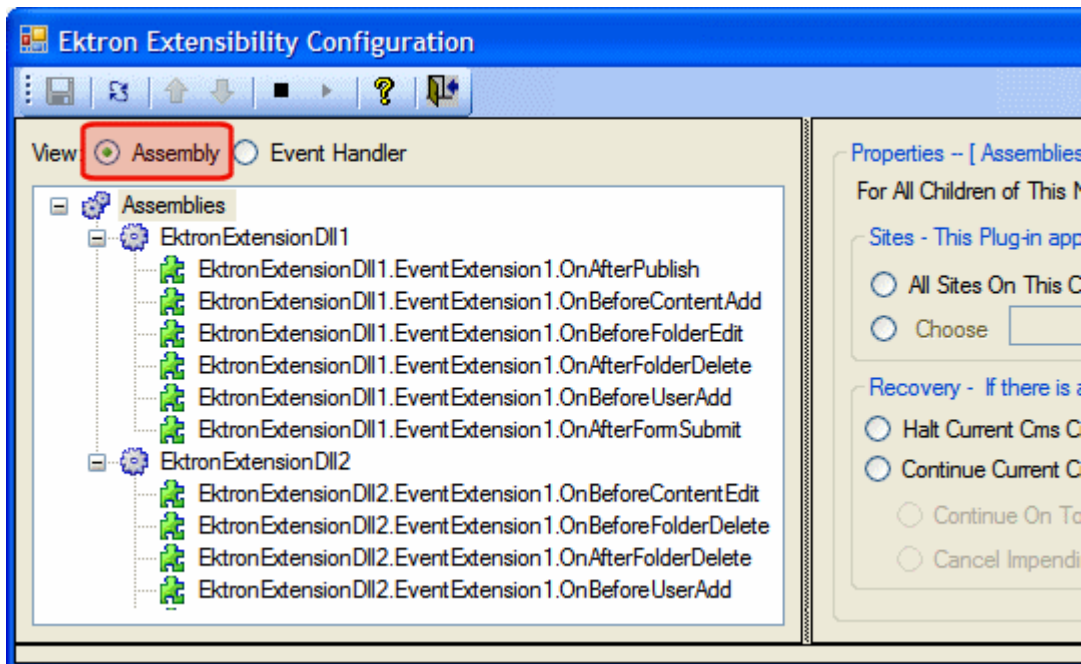
Opening the Extensibility Configuration Editor

To view Plug-in Extensions in the Extensibility Configuration Editor, click **Start > Programs > CMS400 > Utilities > Plug-in Configuration** on your CMS400.NET server. You can also access the editor using the `ConfigurationUtility.exe` file located in: `C:\Program Files\Ektron\CMS400SDK`.

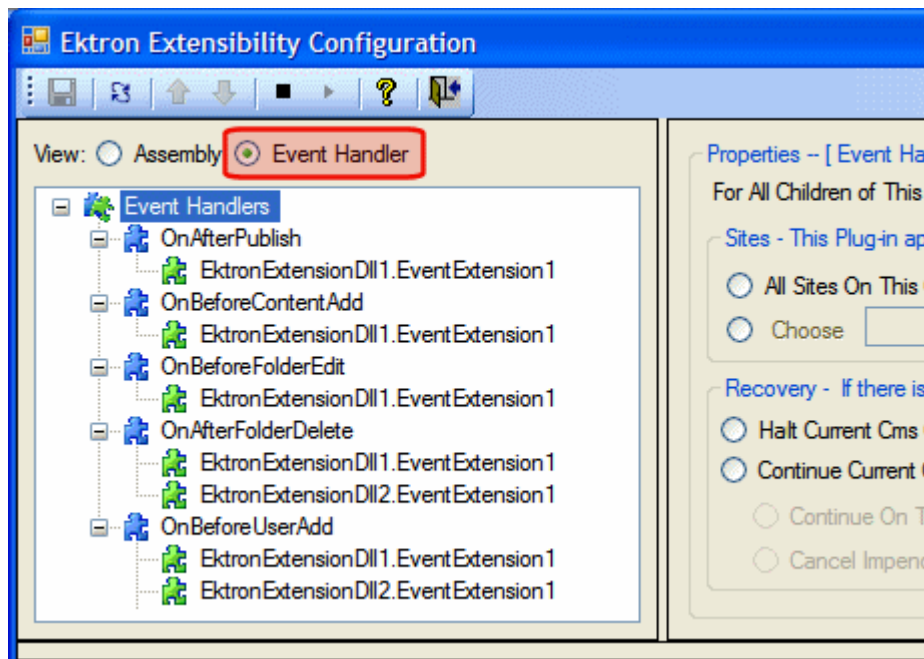
Viewing Plug-in Extensions in the Extensibility Configuration Editor

The Extension Plug-ins in the Extensibility Configuration Editor are displayed in a Tree format that enables you to show or hide each branch of the tree.

You can view the plug-in Extensions by Assembly or by Event Handler. When viewing by Assembly, you see each Plug-in Extension with all of the events for that extension nested below it.







When viewing by Event Handler, you see each event with all of the Plug-in Extensions it belongs to nested below it.



The Extensibility Configuration Editor's Toolbar

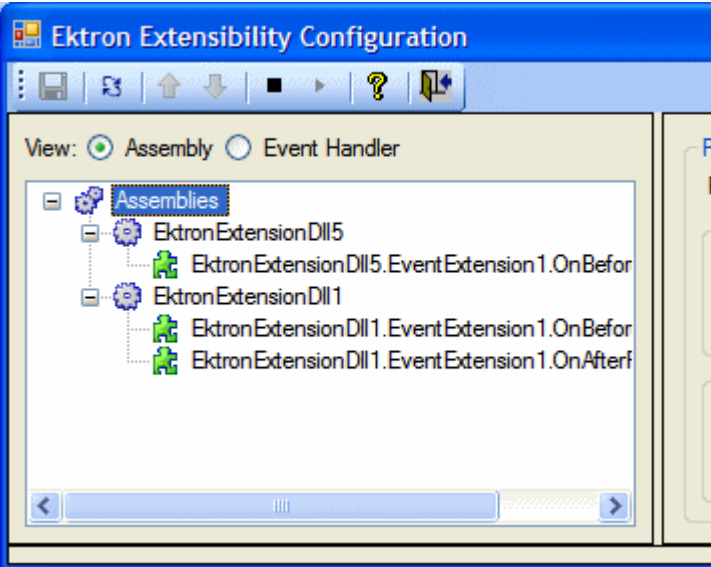
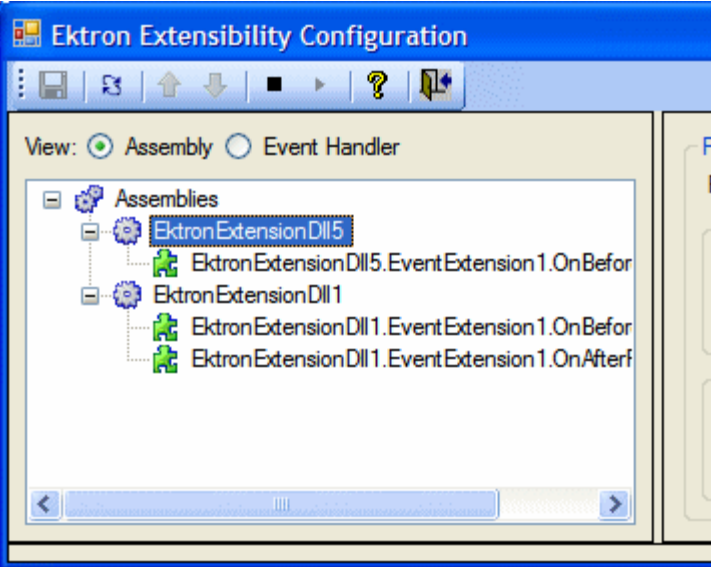
The plug-in toolbar allows you to Save settings, Refresh, Start and Stop, and Reorder plug-ins and events.

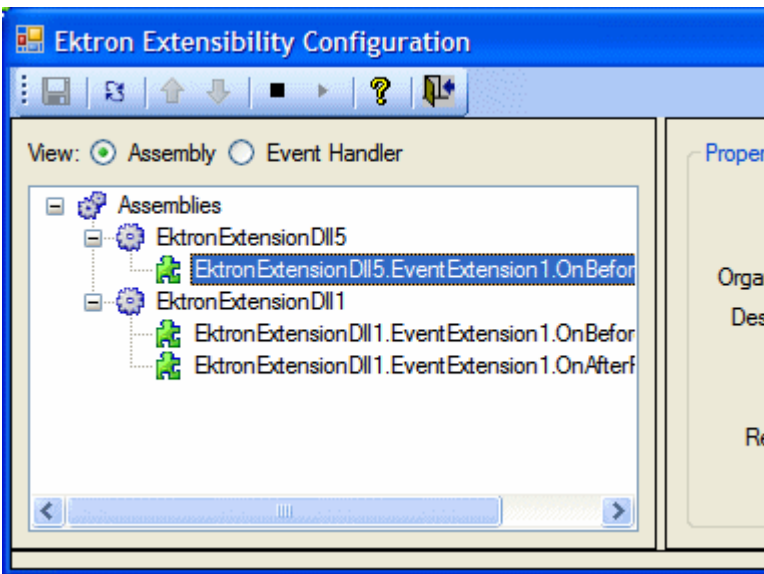
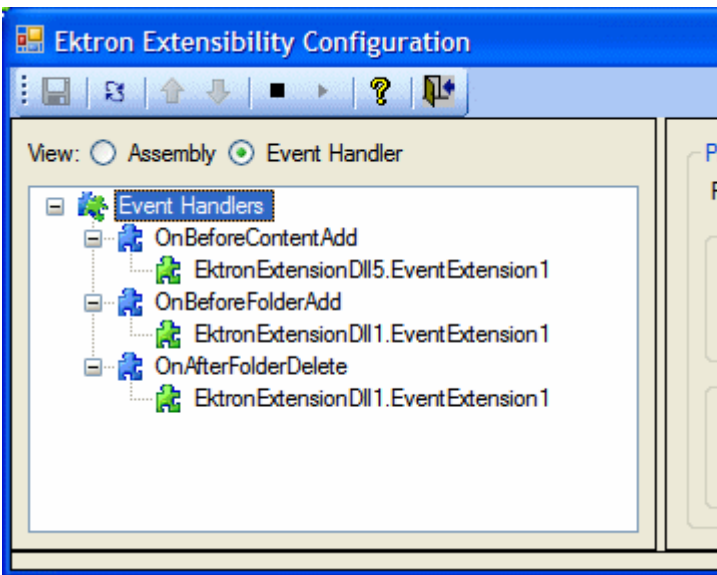
The table below shows each toolbar button and its description.

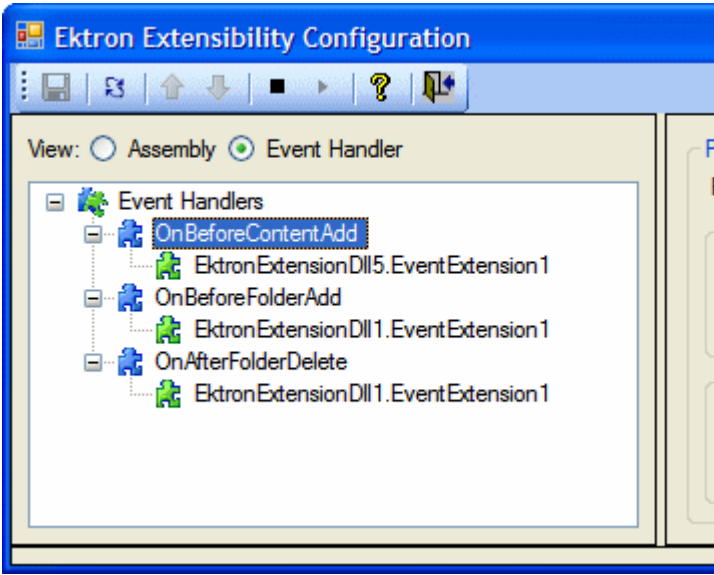
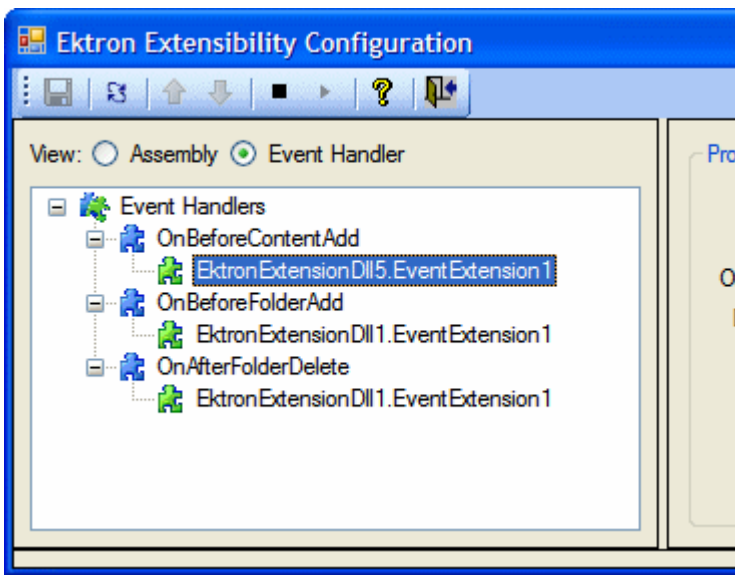
Command	Button	Description	See Also
Save		Saves the changes made in the editor.	
Refresh		Refreshes the editor and displays any new plug-ins that are added to the Extensions folder.	
Move Event Up		Moves an event to a higher priority when multiple events of the same type are used.	"Setting the Plug-in Order of Execution by Event" on page 835
Move Event Down		Moves an event to a lower priority when multiple events of the same type are used.	"Setting the Plug-in Order of Execution by Event" on page 835
Disable Selected Event		Stops the event or plug-in from running.	"Enabling and Disabling Plug-ins and Events" on page 834
Enable Selected Event		Starts the event or plug-in.	"Enabling and Disabling Plug-ins and Events" on page 834
Help		Launches help information.	
Close Editor		Exits the editor.	

Setting Plug-in Extension and Event Properties

When using the Assembly view, you can set properties at the Assemblies level, the Plug-in Extension level or at the event level. When using the Event Handler view, you can set properties at the Event Handler level, the Event level or the Plug-in Extension level. The table below explains the different levels.

Level	Where the Level Appears in the Editor
<p>Assembly View</p> <p>Assemblies - All plug-ins and events below this level use the properties applied to this level.</p>	 <p>The screenshot shows the 'Ektron Extensibility Configuration' dialog box with the 'Assembly' view selected. The 'Assemblies' tree is expanded, showing 'EktronExtensionDll5' and 'EktronExtensionDll1'. Under 'EktronExtensionDll5', there is an event extension 'EktronExtensionDll5.EventExtension1.OnBefore'. Under 'EktronExtensionDll1', there are two event extensions: 'EktronExtensionDll1.EventExtension1.OnBefore' and 'EktronExtensionDll1.EventExtension1.OnAfter'.</p>
<p>Plug-in Extension - The plug-in you choose and all the events associated with that plug-in use the same properties.</p>	 <p>The screenshot shows the 'Ektron Extensibility Configuration' dialog box with the 'Assembly' view selected. The 'Assemblies' tree is expanded, and 'EktronExtensionDll5' is selected. Under 'EktronExtensionDll5', there is an event extension 'EktronExtensionDll5.EventExtension1.OnBefore'. Under 'EktronExtensionDll1', there are two event extensions: 'EktronExtensionDll1.EventExtension1.OnBefore' and 'EktronExtensionDll1.EventExtension1.OnAfter'.</p>

Level	Where the Level Appears in the Editor
<p>Event - Properties are assigned to each individual event.</p>	 <p>The screenshot shows the 'Ektron Extensibility Configuration' dialog with the 'View' set to 'Assembly'. The 'Assemblies' tree is expanded, showing 'EktronExtensionDll5' and 'EktronExtensionDll1'. Under 'EktronExtensionDll5', the event 'EktronExtensionDll5.EventExtension1.OnBefore' is selected.</p>
<p>Event Handler View</p>	
<p>Event Handler - All events and plug-ins below this level use the properties applied to this level.</p>	 <p>The screenshot shows the 'Ektron Extensibility Configuration' dialog with the 'View' set to 'Event Handler'. The 'Event Handlers' tree is expanded, showing 'OnBeforeContentAdd', 'OnBeforeFolderAdd', and 'OnAfterFolderDelete'. Each event has a corresponding 'EktronExtensionDll1.EventExtension1' plug-in.</p>

Level	Where the Level Appears in the Editor
<p>Event - The event you choose and all the plug-ins associated with that event use the same properties.</p>	 <p>The screenshot shows the 'Ektron Extensibility Configuration' window with the 'Event Handler' view selected. Under the 'Event Handlers' category, 'OnBeforeContentAdd' is highlighted. Below it, three plug-ins are listed: 'EktronExtensionDll5.EventExtension1', 'EktronExtensionDll1.EventExtension1', and 'EktronExtensionDll1.EventExtension1'.</p>
<p>Plug-in Extension - Properties are assigned to each plug-in.</p>	 <p>This screenshot is similar to the one above, but the 'EktronExtensionDll5.EventExtension1' plug-in under the 'OnBeforeContentAdd' event is now highlighted, illustrating the 'Plug-in Extension' level.</p>

Property List

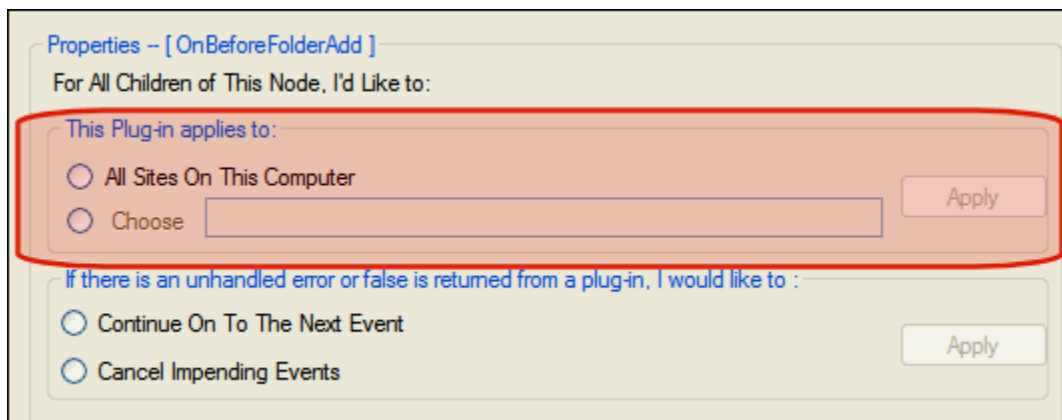
The following table lists the properties contained in the Extensibility Configuration Editor.

Property	Description	See Also
Sites This Plug-in Applies to:	Choose the sites to which this plug-in or event applies. You need only choose a site if you are using Multi-site support.	"Applying Plug-ins and Events to Specific Sites" on page 830 and The Administrator Manual section "Managing Your CMS400 System" > "Support for Multiple Sites"
Recovery If there is an unhandled error or false is returned from a plug-in, I would like to:	Choose what happens when there is an unhandled error.	"Determining What Happens After an Error Occurs" on page 832
Name	The name of the plug-in or event is displayed. This property is set in the plug-in's code.	"Setting the Information Properties in the Plug-in Code" on page 833
Author	The author of the plug-in or event. This property is set in the plug-in's code.	"Setting the Information Properties in the Plug-in Code" on page 833
Organization	The organization or company that created the plug-in or event. This property is set in the Plug-in's code.	"Setting the Information Properties in the Plug-in Code" on page 833
Status	Decide whether plug-ins and events are enabled or disable.	"Enabling and Disabling Plug-ins and Events" on page 834
Order	Decide the order of execution for events of the same type that appear in multiple plug-ins. For example, if the OnBeforeContentAdd event appears in EktronExtDLL5.EventExt1 and EktronExtDLL2.EventExt1, you can choose which plug-in's event is executed first.	"Setting the Plug-in Order of Execution by Event" on page 835

Applying Plug-ins and Events to Specific Sites

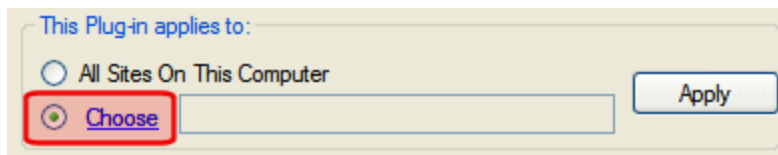
If you are using the Multi-site Support feature in CMS400.NET, you may find it necessary to apply a plug-in or event to one site and not another. To accomplish this, use the Sites property. It allows you to assign events and plug-ins to all the

sites on a server or to select sites on the server. For example, you are appending HTML content by applying a company's copyright information to a site and you are supporting multiple sites with CMS400.NET.

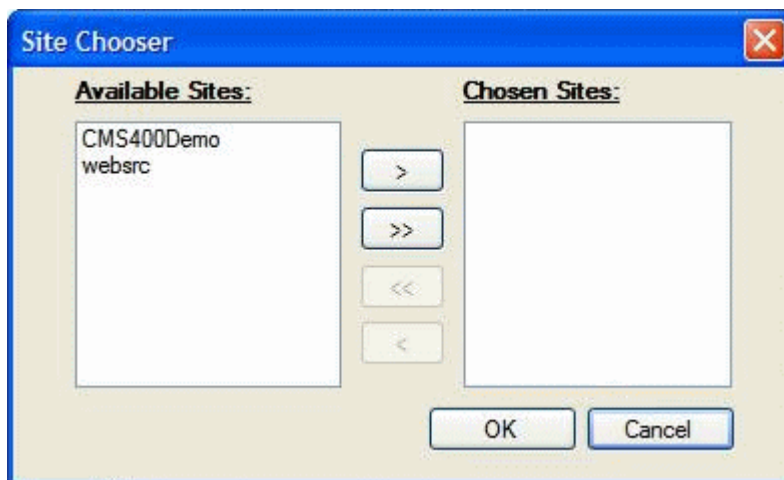


To apply a plug-in or event to a specific site, follow these steps.

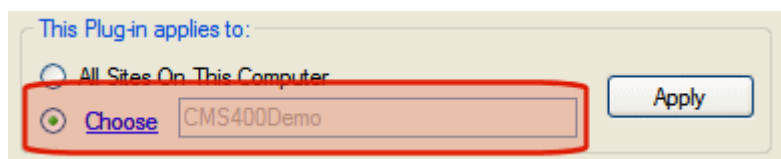
1. Open the Ektron Extensibility Configuration Editor by clicking **Start > Programs > CMS400 > Utilities > Plug-in Configuration**.
2. Click the plug-in or event in the menu tree.
3. Under the Sites property, select either **All Sites on This Computer** or **Choose** radio buttons. The first time you select the Choose radio button, the Site Chooser screen appears.




NOTE If the Site Chooser screen does not appear when you click the **Choose** radio button, click the **Choose** hyperlink next to the radio button.



4. Highlight the sites in the **Available Sites** column to which the plug-in or event will be assigned.
5. Click the > button to move the selected sites to the **Chosen Sites** column. Click >> to move all of the sites. To remove a selection from the **Chosen Sites** column, highlight it and click the < button. To remove all selections, click the << button.
6. Click **OK**.
7. The chosen sites appear in the text box.



8. Click **Apply**.
9. Click the Save () button.


Determining What Happens After an Error Occurs

The Recovery property allows you to decide how to handle errors that arise when utilizing plug-ins and events. For this property there are two choices:

- **Halt Current CMS Operation** - in the event of an unhandled error, this selection stops the current operation completely. For example, if you publish a piece of content and the OnBeforePublish event has an unhandled error, the content is not published and the error message associated with the action is displayed.
- **Continue Current CMS Operation** - when this option is selected, the choices below are available.

- **Continue On to the Next Event** - in the event of an unhandled error, the process moves to the next event. For example, if you had three plug-ins that contained the OnBeforePublish event and it failed in the first instance, the process would then move on to the next OnBeforePublish event.
- **Cancel Impending Events** - in the event of an unhandled error, the process cancels all impending occurrences of the event. For example, if you had three plug-ins that contained the OnBeforePublish event and it failed in the first instance, all subsequent OnBeforePublish events are canceled.

To apply the Recovery property to a Plug-in or Event, follow these steps.

1. Open the Ektron Extensibility Configuration Editor by clicking **Start > Programs > CMS400 > Utilities > Plug-in Configuration**.
2. Click the plug-in or event in the menu tree.
3. Under the Recovery property, select the **Halt Current CMS Operation** or **Continue Current CMS Operation** radio button.
4. If you select Continue Current CMS Operation, select **Continue On to the Next Event** or **Cancel Impending Events** radio button. Otherwise, continue to the next step.
5. Click **Apply**.
6. Click the Save () button.

Setting the Information Properties in the Plug-in Code

You can add Author, Organization and Description information to your plug-in code. This helps you identify who wrote the plug-in and what it does. This information can be assigned at the assembly level (the plug-in) or the method level (the event). The method level attributes override the attributes assigned at the assembly level.



You assign assembly level attributes in the AssemblyInfo.cs or AssemblyInfo.vb depending on the language you are using. Method level attributes are assigned in the plug-in code above the method. The example below is at the method level using C#:

```
[ExtensionAuthor("John Smith")]
[ExtensionDescription("Appends text to the html")]
public override bool OnBeforePublish()
```

The following table shows more syntax examples for C# and VB.

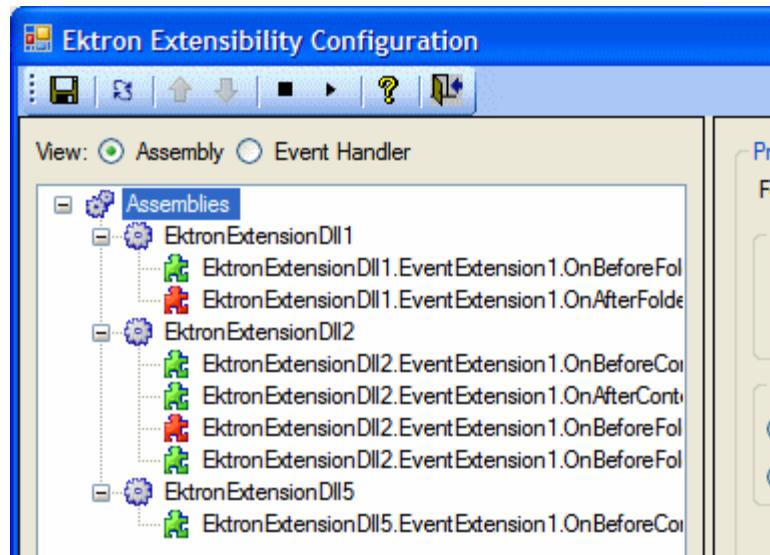
Example Type	Language	Example
Method Level	C#	<pre>[ExtensionAuthor("John Smith")] [ExtensionDescription("Appends text to the html")] public override bool OnBeforePublish()</pre>
Assembly Level	C#	<pre>[assembly: AssemblyDescription("Assembly Level Description")] [assembly: Ektron.Cms.Extensibility.ExtensionAuthor("John Smith")] [assembly: AssemblyCompany("Ektron Company Extension")]</pre>
Method Level	VB	<pre><ExtensionAuthor("John Smith")> _ <ExtensionDescription("Before Content Add Description")> _ Public Overrides Function OnBeforeContentAdd() As Boolean</pre> <hr/> <p>Note: The continue character “_” must be used at the end of each description line.</p> <hr/>
Assembly Level	VB	<pre><Assembly: AssemblyDescription("Assembly Level Description")> <Assembly: Ektron.Cms.Extensibility.ExtensionAuthor("John Smith")> <Assembly: AssemblyCompany("Ektron Company Extension")></pre>

Enabling and Disabling Plug-ins and Events

Plug-ins and events are enabled and disabled using the Enable Events () and Disable Events () buttons. You can enable or disable single events by clicking on the event and then clicking the Enable or Disable button.

If you want to enable or disable all the events in a plug-in, click on the plug-in while in the Assemblies view. Next, click the **Enable** or **Disable** button. To enable or disable all of the events of the same type, click on the event while in the Event Handler view. Next, click the **Enable** or **Disable** button.

When events are enabled, their puzzle icon is green. If they are disabled, their puzzle icon is red. Plug-ins and events are enabled by default.



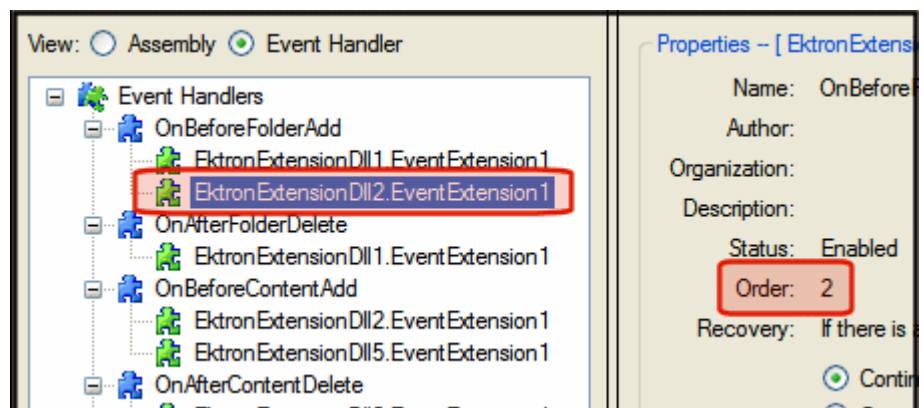
Setting the Plug-in Order of Execution by Event

Setting the plug-in execution order allows you to control when events of the same type are executed. For example, if you have three OnBeforePublish events in three separate plug-ins, you can decide which OnBeforePublish event happens first, second and third.

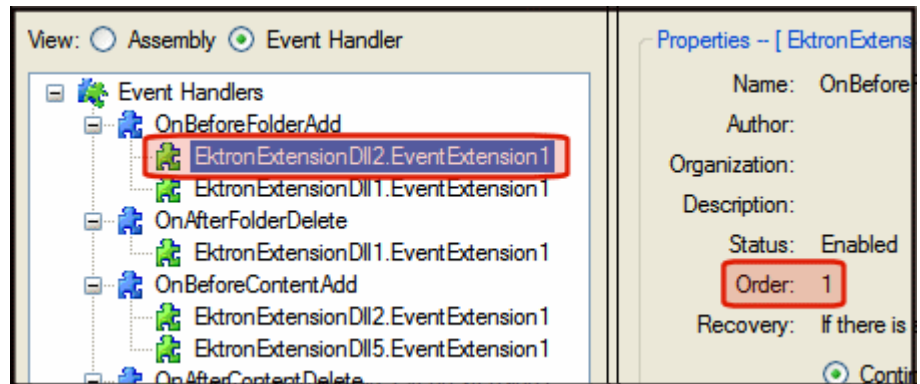
The order of events can only be set in Event Handler view. This way you see the plug-ins sorted by event. To change the plug-in execution order, click a plug-in and use the Event Up (↑) or Event Down (↓) button to reorder plug-ins.

In the before and after example below, the EktronExtensionDll2.EventExtension1 has been move to order position 1 (one) for the OnBeforeFolderAdd event.

Before:



After:



Connecting to the Web Service API From a Plug-in

You can connect to the Web Service API from a plug-in by using a Web service proxy. Calling a Web service proxy accessor creates a connection to a particular Web service. For example, using the `GetContentProxy()` creates a connection to the Web Service API `Content.asmx`.

When using the Web service proxy, the Web service connects to the site that produced the event. For example, if you have three sites and site two uses a plug-in with the `OnBeforePublish` event that calls a Web service, the Web service connects back to site two only.

Because the Plug-in Architecture and CMS400.NET run on the same server, no authentication is needed when making a Web Service API call. Once a connection is made, you can call any Web service method that is available. For information on Web service methods, click **Help > Contents > Ektron CMS400.NET API Documentation > Ektron Namespace > Web Services API**.

Below is a list of available Web service proxies.

NOTE In the Web Service Connection column, the ... refers to </<site webroot>/Workarea/webservices/WebServiceAPI>

Proxy Accessor	Returns	Web Service Connection
<code>GetActiveDirectoryUserProxy()</code>	ActiveDirectoryUserProxy	.../User/ActiveDirectoryUser.asxm
<code>GetAssetProxy()</code>	AssetProxy	.../Content/Assest.asmx
<code>GetBlogProxy()</code>	BlogProxy	.../Content/Blog.asmx
<code>GetBusinessRulesProxy()</code>	BusinessRulesProxy	.../BusinessRules.asmx

Proxy Accessor	Returns	Web Service Connection
GetCalendarEventTypeProxy()	CalendarEventTypeProxy	.../Calendar/CalendarEventType.aspx
GetCalendarProxy()	CalendarProxy	.../Calendar/Calendar.aspx
GetContentProxy()	ContentProxy	.../Content/Content.aspx
GetCustomFieldsProxy()	CustomFieldsProxy	.../CustomFields.aspx
GetFolderProxy()	FolderProxy	.../Folder.aspx
GetFontProxy()	FontProxy	.../Font.aspx
GetFormProxy()	FormProxy	.../Content/Form.aspx
GetLibraryProxy()	LibraryProxy	.../Library.aspx
GetMetadataProxy()	MetadataProxy	.../Metadata.aspx
GetPermissionsProxy ()	PermissionsProxy	.../Permissions.aspx
GetSearchManagerProxy()	SearchManagerProxy	.../Search/SearchManager.aspx
GetSiteMapProxy()	SiteMapProxy	.../SiteMap.aspx
GetSiteProxy()	SiteProxy	.../Site.aspx
GetTaskProxy()	TaskProxy	.../Task.aspx
GetTaskCategoryProxy()	TaskCategoryProxy	.../Task/TaskCategory.aspx
GetTaskCategoryTypeProxy()	TaskCategoryTypeProxy	.../Task/TaskCategoryType.aspx
GetThreadedDiscussionProxy()	ThreadedDiscussionProxy	.../Content/ThreadedDiscussion.aspx
GetUserProxy()	UserProxy	.../User/User.aspx

Debugging a Plug-in

When a Plug-in Extension is not working, it is necessary to debug it. The sections explain how to debug a C# plug-in or a VB plug-in.

- ["Debugging a C# Plug-in Extension" on page 838](#)
- ["Debugging a VB Plug-in Extension" on page 847](#)

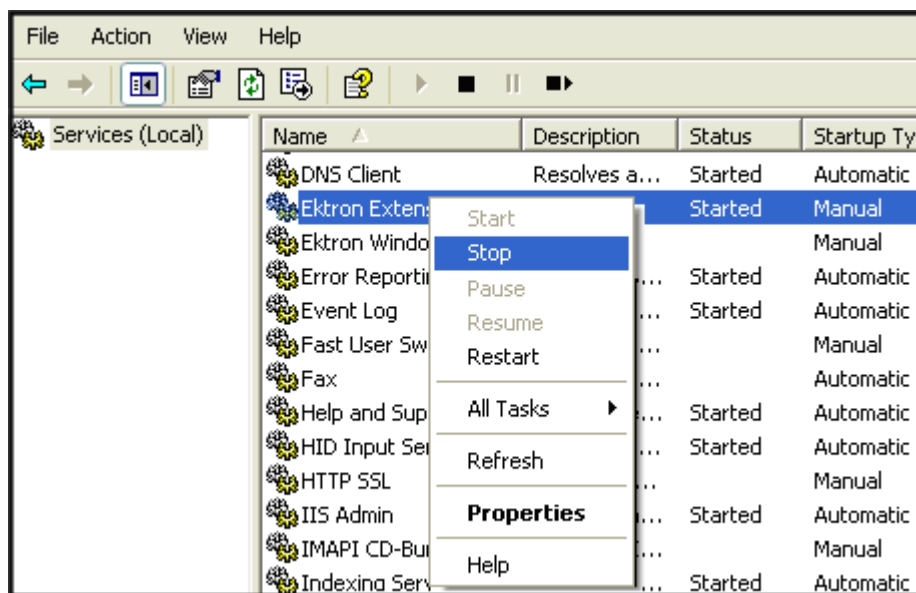
NOTE You should only debug a plug-in on a non-production machine. Real-time plug-in updates are disabled when debugging takes place. For example, deleting or replacing a plug-in that's in the watch directory is not supported while in debug mode.

Debugging a C# Plug-in Extension

Steps

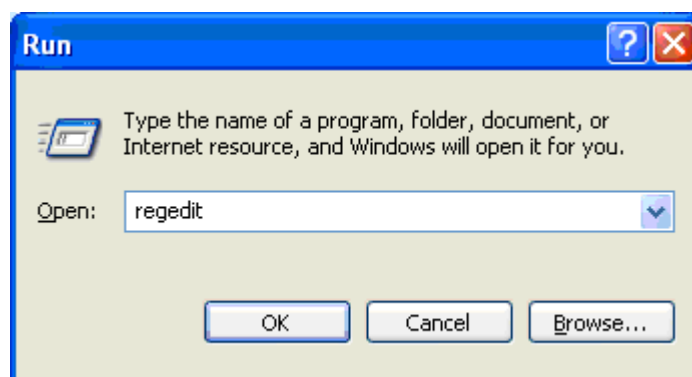
1. Navigate to **Start > Control Panel > Administrative Tools > Services** and stop the Ektron Extensibility Server.

Important: To have the debug process automatically start and stop the Ektron Extensibility Server, see the Note in step five.

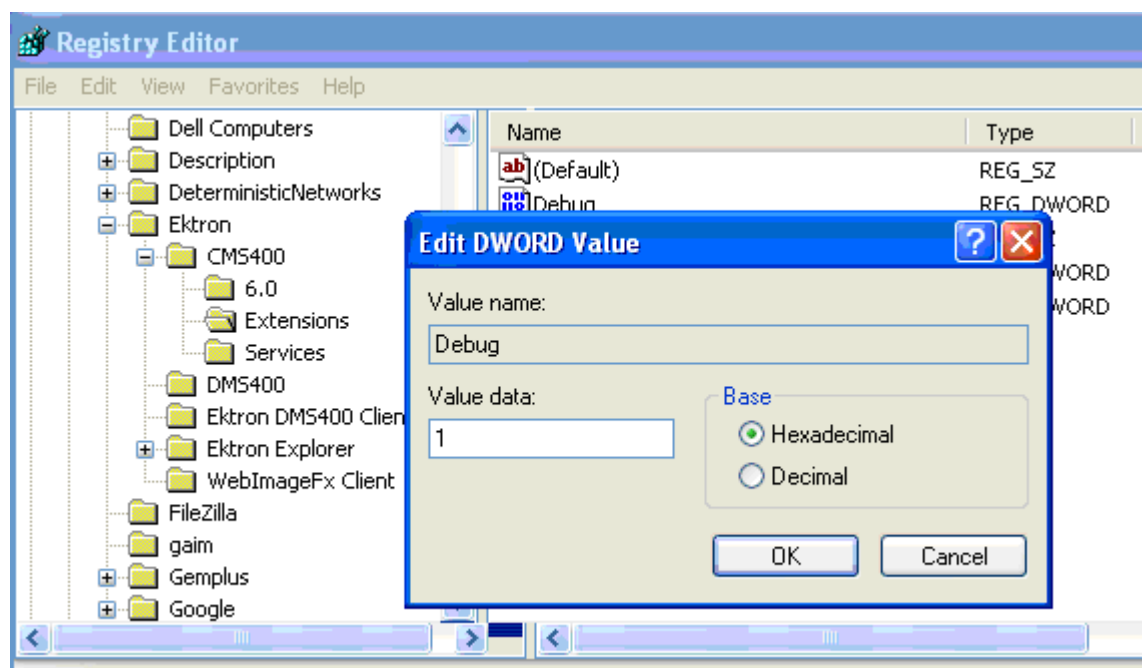


Steps

2. Run Regedit by clicking **Start > Run** and typing **regedit** in the Open text box. Next, click **OK**.

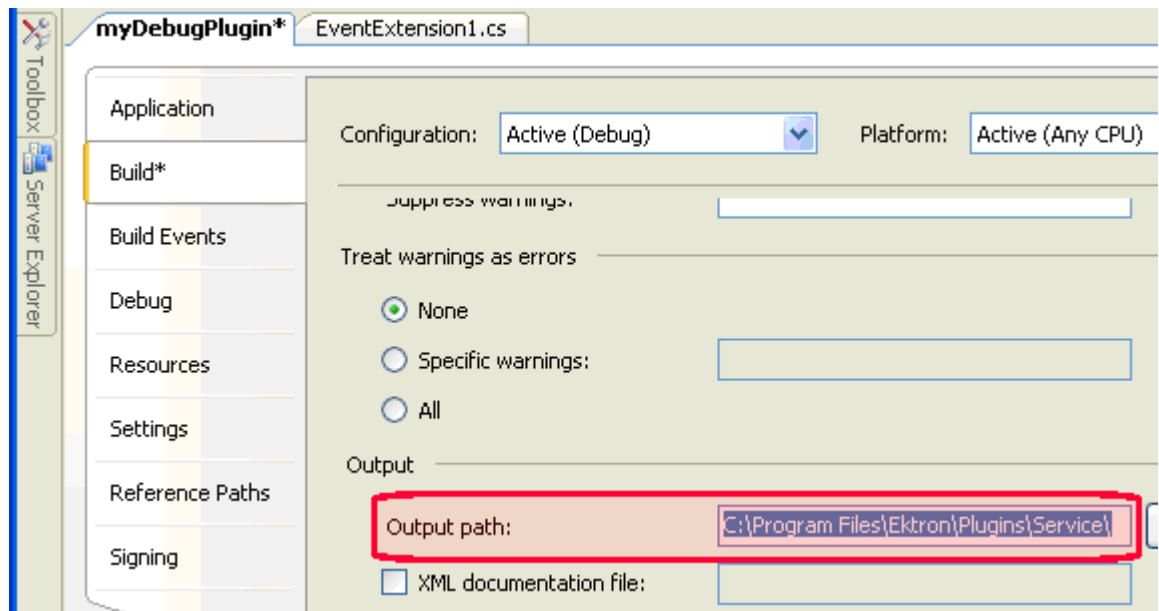


3. Change the entry **HKEY_LOCAL_MACHINE\SOFTWARE\Ektron\CMS400\Extensions\Debug** to 1. Use either Hexadecimal or Decimal.



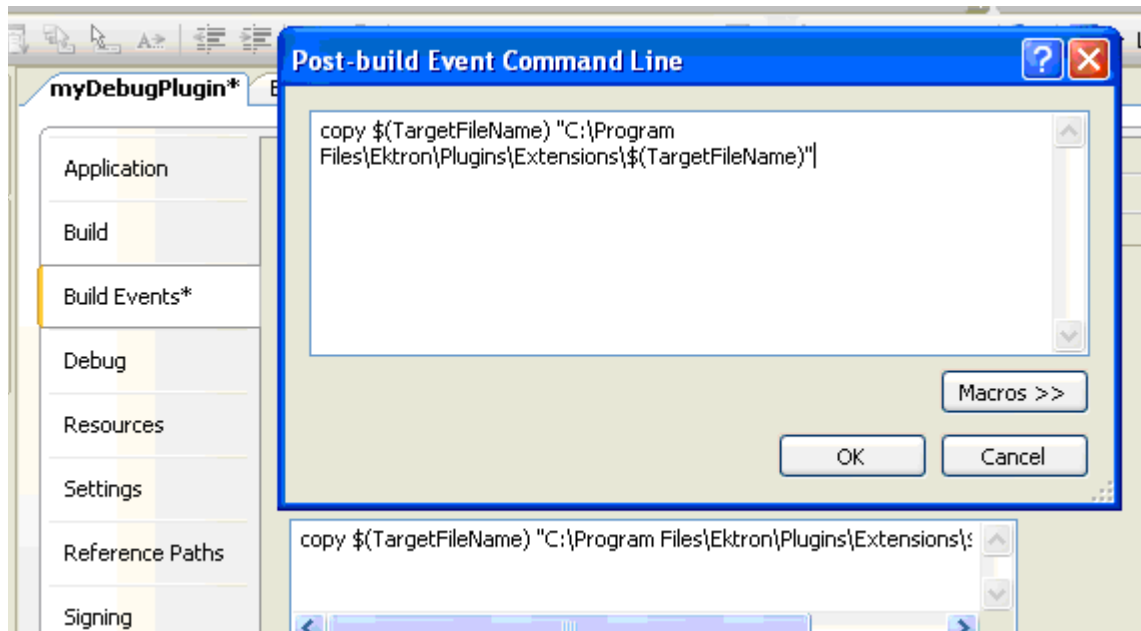
Steps

4. In Visual Studio 2005, right click on the project in the Solution Explorer and select properties. Next, click the Build tab change the **Output path** of the plug-in so it builds in the service directory. This is the directory that has the ExtensionService.exe in it. The default install location is C:\Program Files\Ektron\Plugins\Service\.

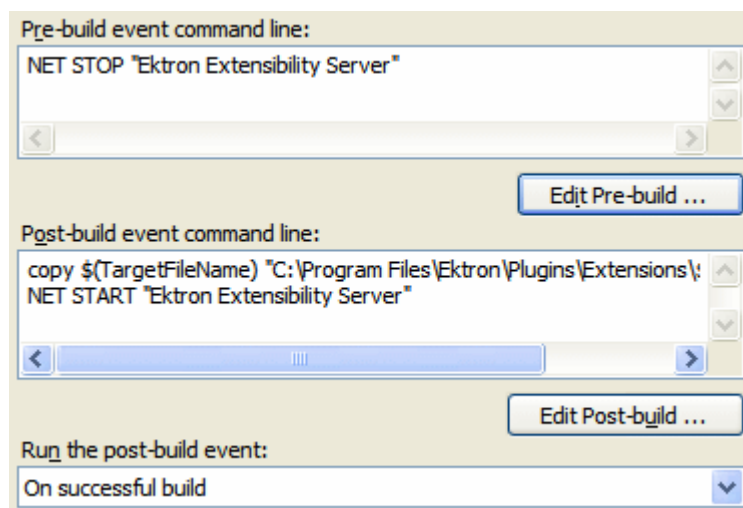


Steps

- Copy the built plug-in to the watch folder via the **Post-build** step on the Build Events settings page. The default install location is `C:\Program Files\Ektron\Plugins\Extensions`.

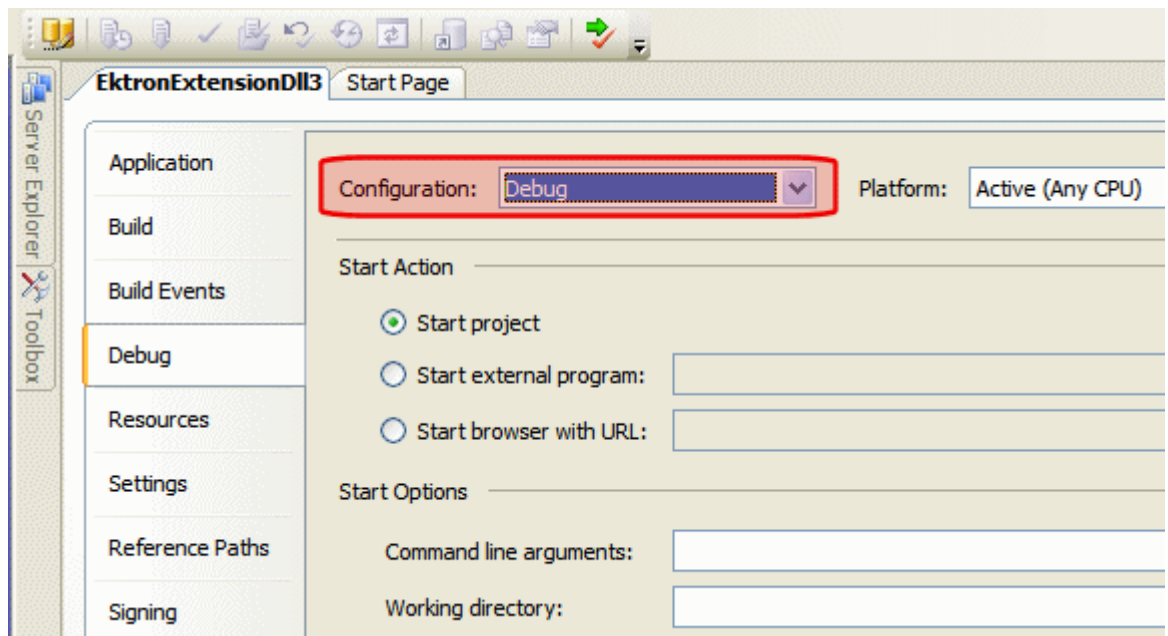


Note: To start and stop the Ektron Extensibility Server automatically, add **NET STOP "Ektron Extensibility Server"** to the Pre-build event and **NET START "Ektron Extensibility Server"** to the Post-build event after the copy command.



Steps

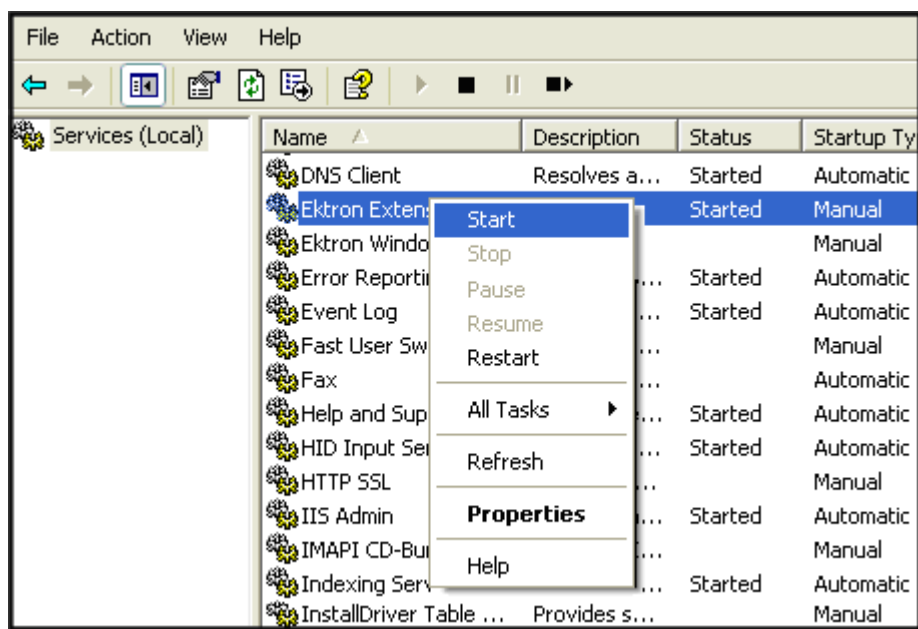
6. On the **Debug** tab, change Configuration to “Debug” and build the plug-in.



Steps

7. Start the Ektron Extensibility server.

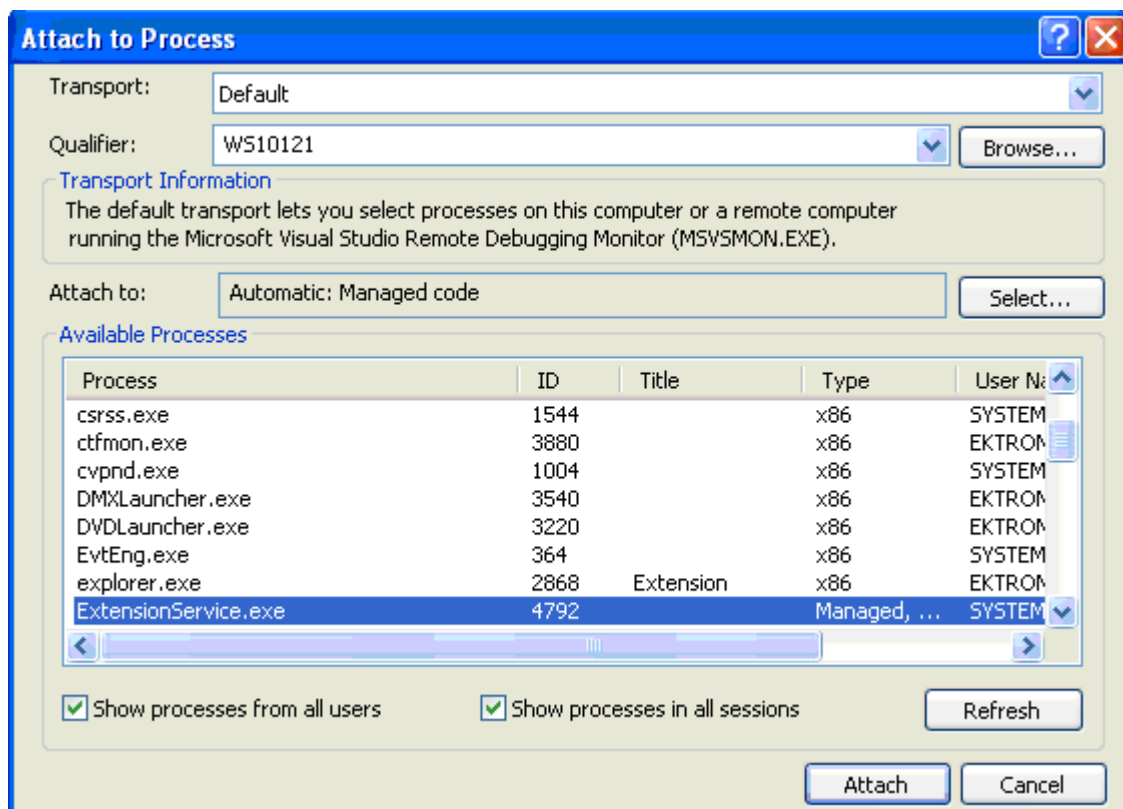
Important: To have the debug process automatically start and stop the Ektron Extensibility Server, see the Note in step five.



Steps

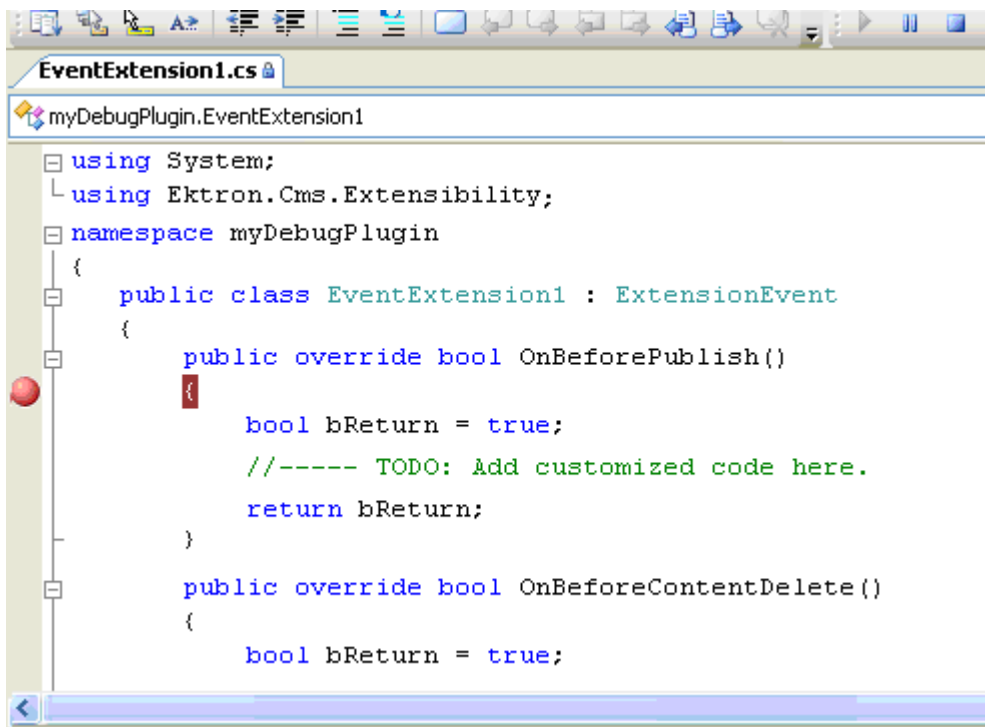
8. In Visual Studio, click **Debug > Attach to Process** or **Ctrl+Shift+R**. Next, click the ExtensionService.exe process, then click the **Attach** button.

Important: If you do not see the ExtensionService.exe, make sure that **Show processes from all users** and **Show processes in all sessions** are checked.



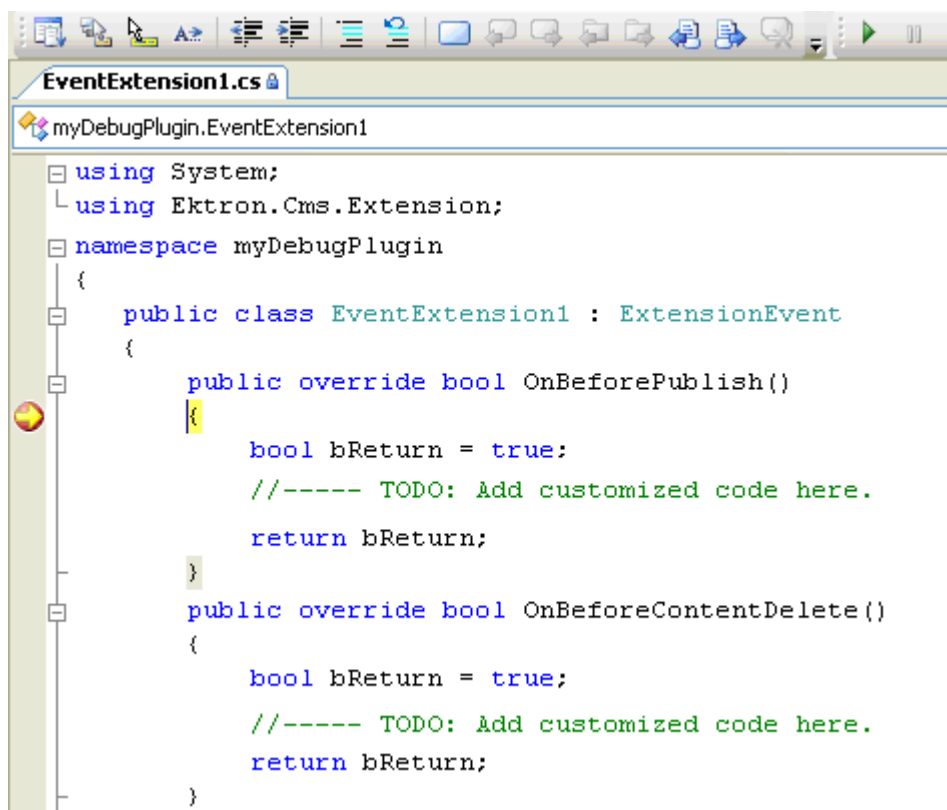
Steps

9. Set the breakpoint(s) in the plug-in where debugging is to take place.



Steps

10. Execute the action within the CMS to trigger the event. For the example breakpoint below, publish a piece of content. The breakpoint is hit as shown below.

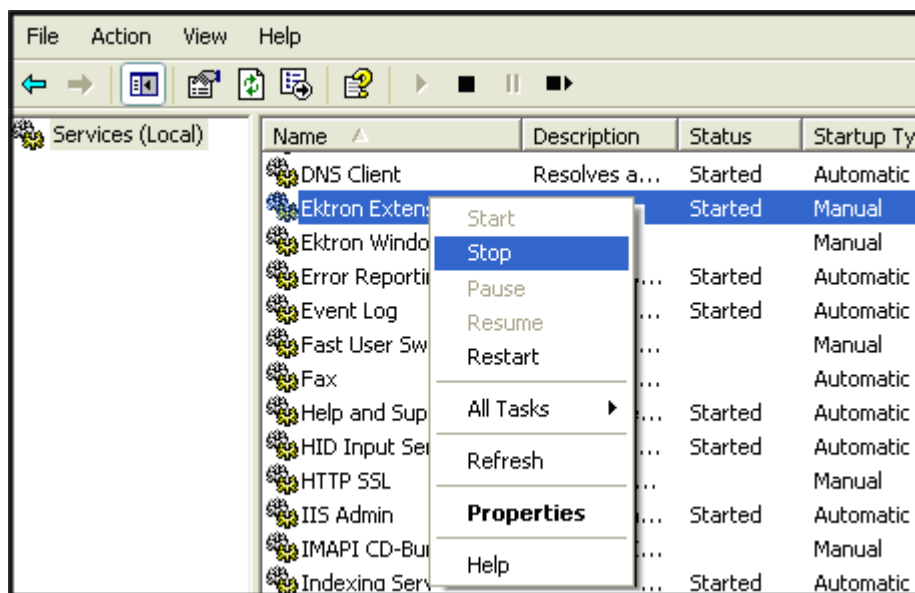


Debugging a VB Plug-in Extension

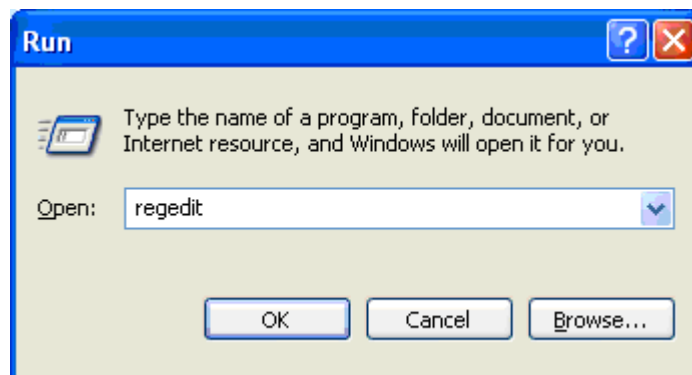
Steps

1. Navigate to **Start > Control Panel > Administrative Tools > Services** and stop the Ektron Extensibility Server.

Important: To have the debug process automatically start and stop the Ektron Extensibility Server, see the Note in step five.

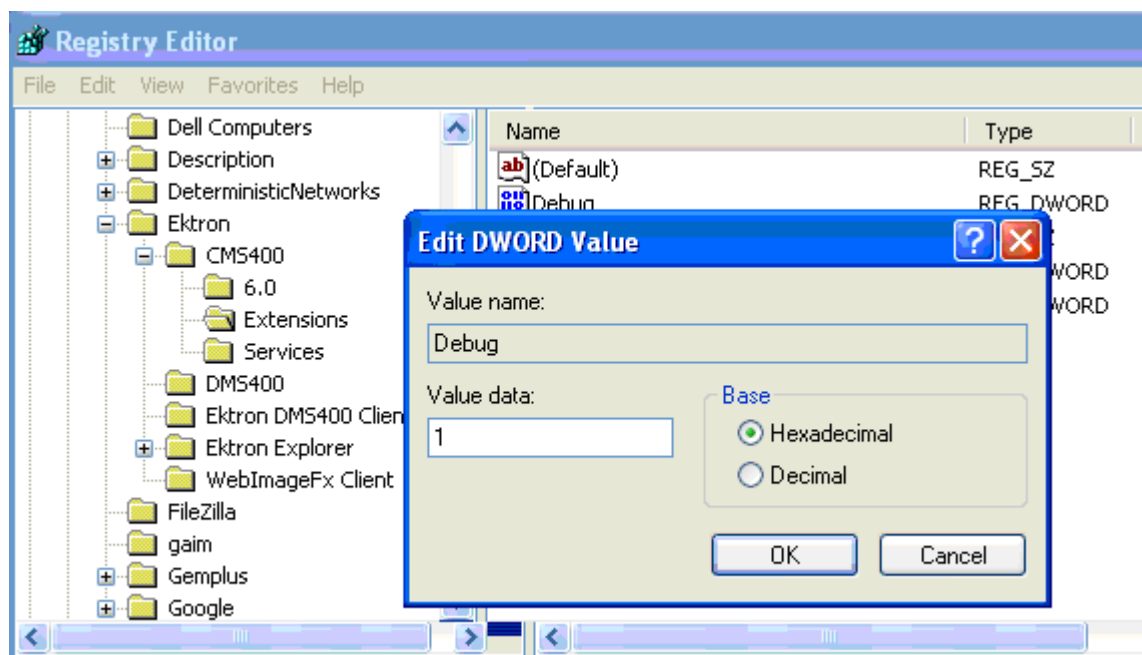


2. Run Regedit by clicking **Start > Run** and typing **regedit** in the Open text box. Next, click **OK**.



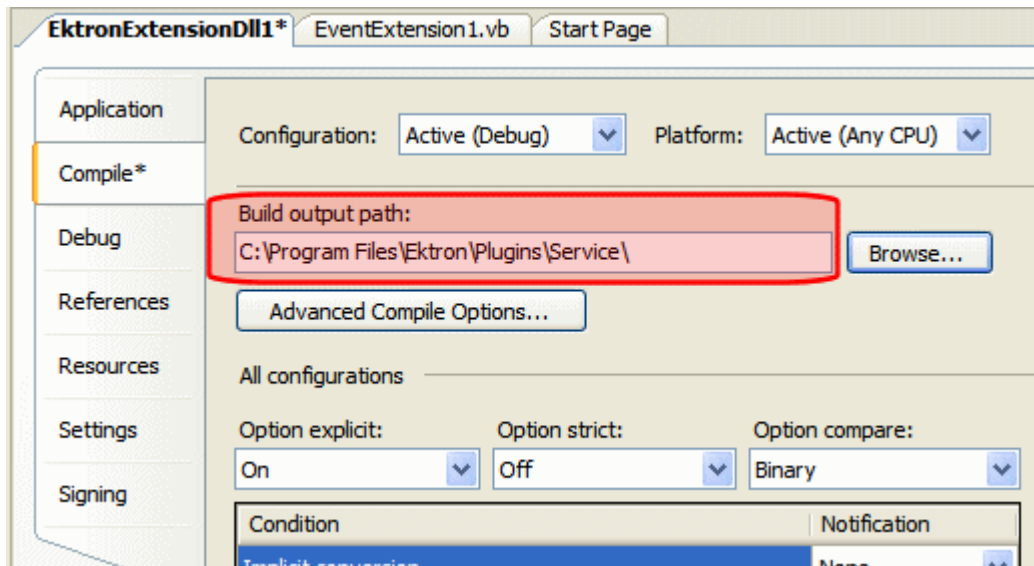
Steps

3. Change the entry `HKEY_LOCAL_MACHINE\SOFTWARE\Ektron\CMS400\Extensions\Debug` to 1. Use either Hexadecimal or Decimal.



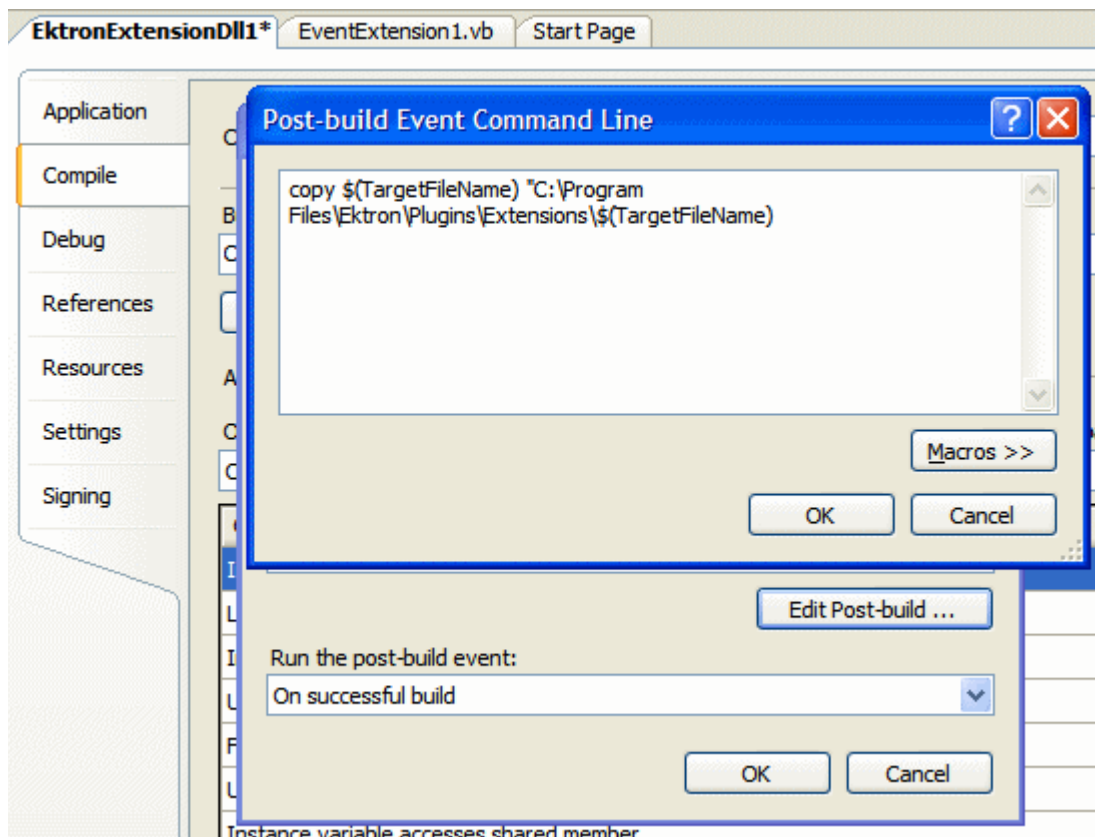
Steps

4. In Visual Studio 2005, right click on the project in the Solution Explorer and select **Properties**. Next, click the **Compile** tab change the **Build Output path** of the plug-in so it builds in the service directory. This is the directory that has the ExtensionService.exe in it. The default install location is C:\Program Files\Ektron\Plugins\Service\.



Steps

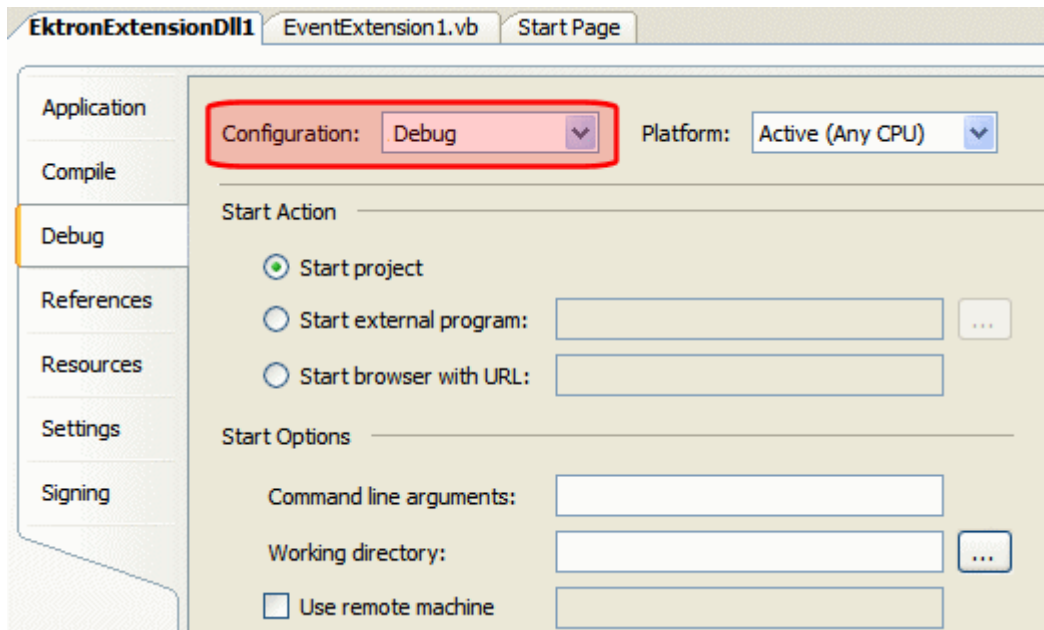
- Copy the built plug-in to the watch folder via the **Post-Build** event. On the Compile tab, click the **Build Events...** button. Next, click the **Edit Post-build...** button. Enter the following information and click **OK**.
`copy $(TargetFileName) "C:\Program Files\Ektron\Plugins\Extensions\$(TargetFileName)"`



Note: To start and stop the Ektron Extensibility Server automatically, add **NET STOP "Ektron Extensibility Server"** to the Pre-build event and **NET START "Ektron Extensibility Server"** to the Post-build event after the copy.

Steps

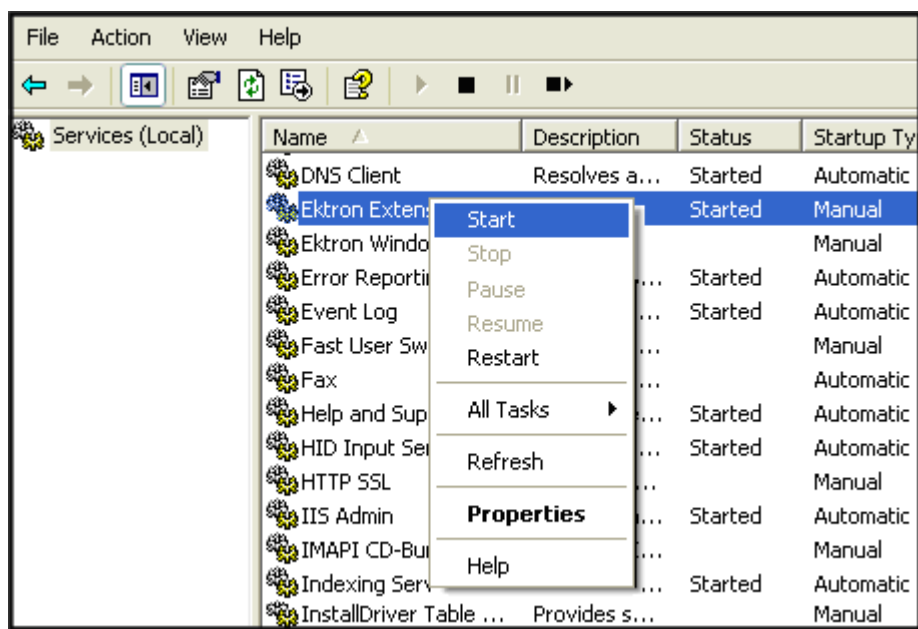
6. On the Debug tab, change **Configuration** to “Debug” and build the plug-in.



Steps

7. Start the Ektron Extensibility Server.

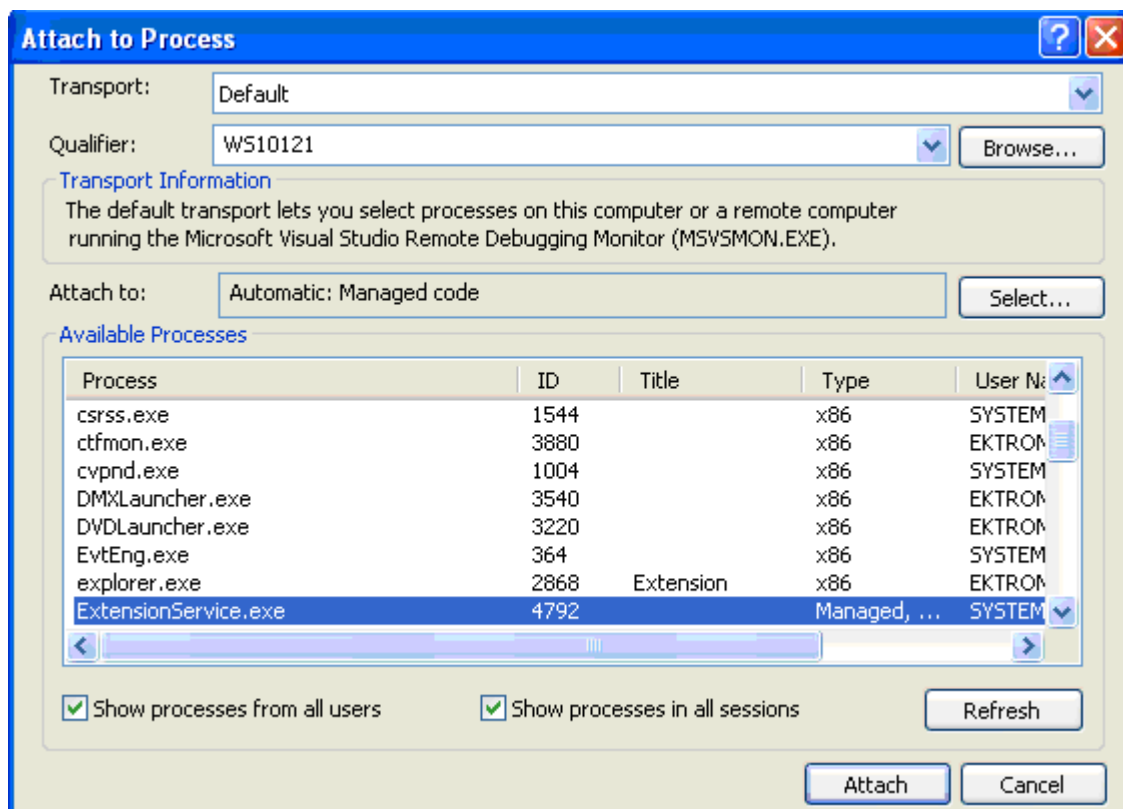
Important: To have the debug process automatically start and stop the Ektron Extensibility Server, see the Note in step five.



Steps

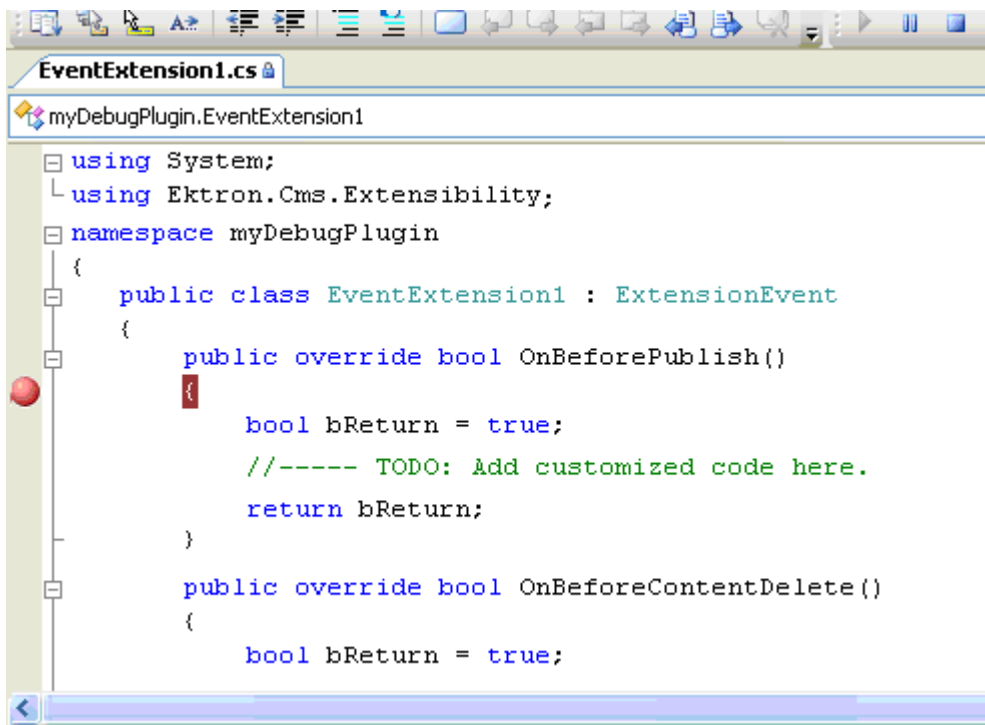
8. In Visual Studio, click **Debug > Attach to Process** or **Ctrl+Shift+R**. Next, click the **ExtensionService.exe** process, then click the **Attach** button.

Important: If you do not see the **ExtensionService.exe**, make sure that **Show processes from all users** and **Show processes in all sessions** are checked.



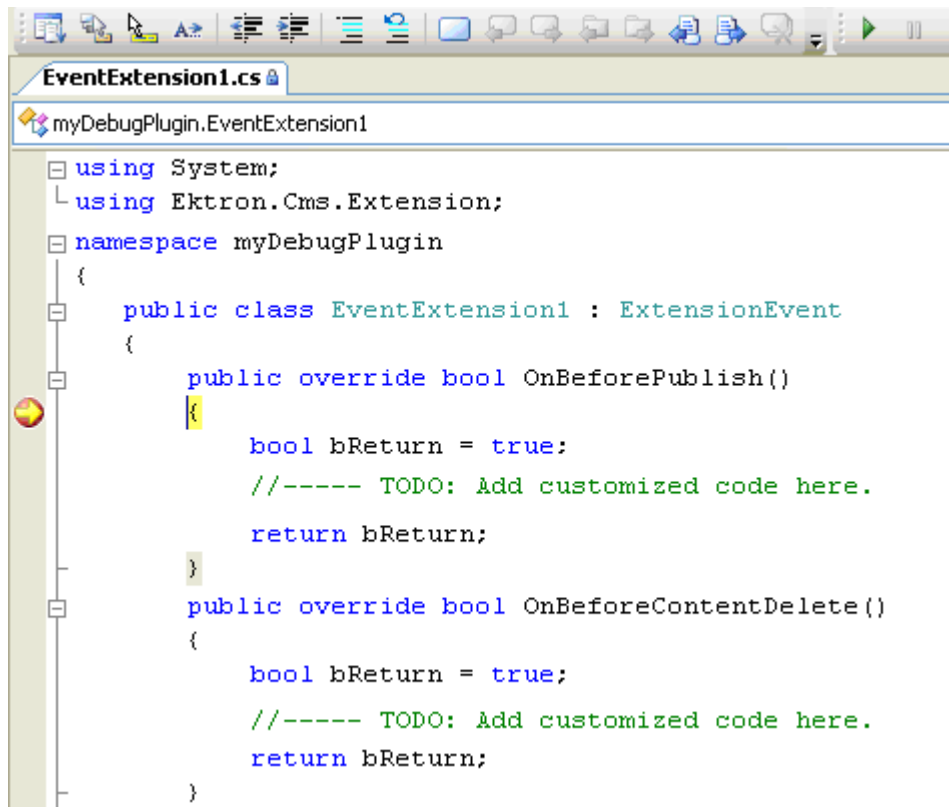
Steps

9. Set the breakpoint(s) in the plug-in where debugging is to take place.



Steps

10. Execute the action within the CMS to trigger the event. For the example breakpoint below, publish a piece of content. The breakpoint is hit as shown below.



Web Services

Introduction to Web Services

Web Services (WS) are reusable software components that can be consumed by variety of clients, independent of operating system, programming language, or Web server, using standard Internet technologies. Because Web Services allow heterogeneous systems to communicate with each other in a standard way, they have received quick acceptance for data interchange over the Web.

It's easy for anyone to integrate your WS into their application. The client application can be a Web application, a Windows application, a Java application, a Visual Studio.net application, or a host of others -- it does not matter as long as it supports Web Services.

A Web Service consists of one or more methods that expose functionality. In this sense, a Web Service is an API that can be accessed using SOAP over HTTP. (SOAP is an XML-based, lightweight protocol that defines the message format of Web Services.)

For example, you might develop a purchasing application that automatically obtains price information from several vendors, lets the user select a vendor, submits the order and then tracks the shipment until it is received. The vendor application, in addition to exposing its services on the Web, might in turn use XML Web services to check a customer's credit, charge the customer's account and set up the shipment with a shipping company.

This manual does not explain how Web Services work. To learn more about them, try these sites.

- http://msdn2.microsoft.com/en-us/library/aa480728.aspx#wsmsplat_topic2
- <http://aspalliance.com/jnuckolls/articles/introws/default.aspx>
- <http://aspnet.4guysfromrolla.com/articles/062602-1.aspx>
- <http://www.15seconds.com/Issue/010430.htm>

CMS400.NET's Implementation of Web Services

CMS400.NET supports XML Web Services. This means that you can create pages that offer your content to any Web-enabled device, and you can create pages that consume existing Web Services. CMS400.NET's Web Services feature can return data separated from presentation. In combination with XML support, this feature lets consumers retrieve your XML data independent of presentation information and act on it however they see fit.

Benefits of Using Web Services

Web Services make it easy to connect with partners. They can also deliver more personal, integrated experiences to users via smart devices, including

PCs. WSs can save time and money by reducing development time, and increase revenue by making your own XML Web services available to others.

WS Components and CMS400.NET Architecture

Web Services require the installation of the .NET Framework on the server.

Ektron's Web Service, ContentWS.asmx, runs within the .NET Framework. Through the Application API class in the Ektron.cms.commonUI.dll, .NET managed code can communicate with asp.net assembly code. ContentWS.asmx utilizes the ektron.com.commonui.dll file to provide access to the Content Business Objects (content blocks, menus, collections, list summaries, etc.).

Software and Hardware Requirements

- .NET framework, version 2.0, must be installed on the server (for more information, see <http://msdn.microsoft.com/netframework/>).
- Ektron CMS400.NET, version 6.0 or higher

Installation of Files

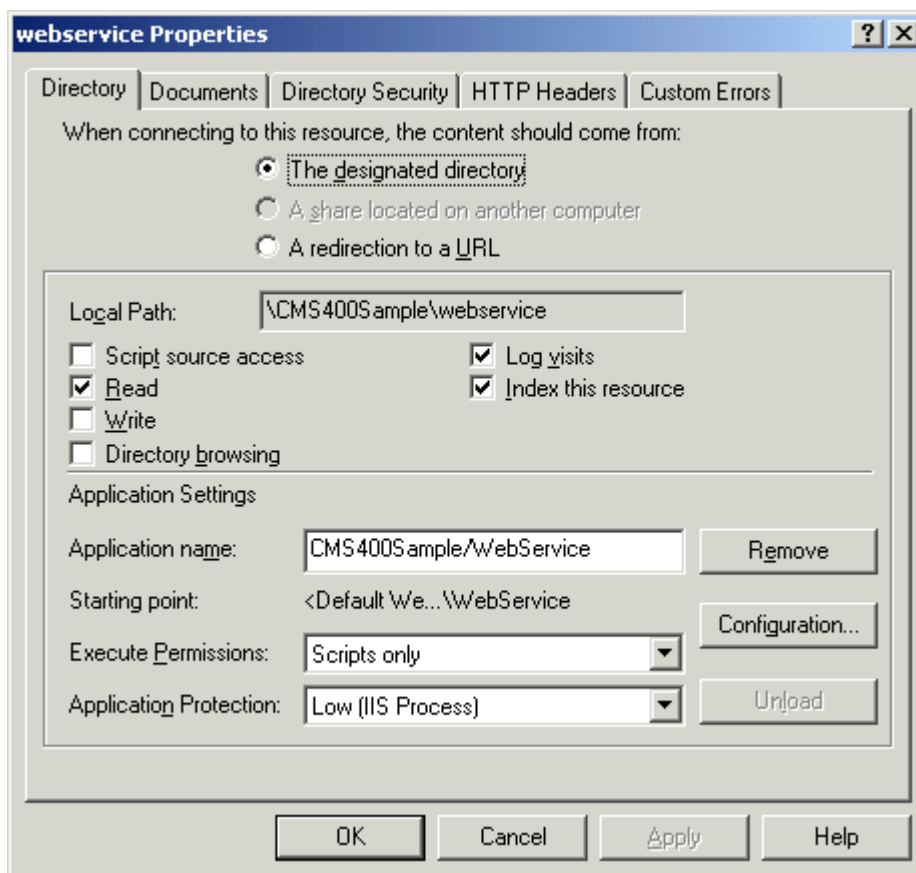
Below is a list of the files and their locations that make up the Web Services feature. They are installed when you install CMS400.NET.

File or folder location	Description
<code>webroot/siteroot/Workarea/webservices/ContentWS.asmx</code>	Entry point for Web Service. This Web page displays examples of all methods.
<code>webroot/siteroot/bin</code>	Directory for compiled Web Service code
<code>webroot/siteroot/bin/WebService.dll</code>	Web Service dll

Setup Instructions

After you install CMS400.NET, you need to open IIS, select *siteroot/workarea*, and make it an application. To make a folder an application, follow these steps.

1. Open IIS.
2. Right click one of the folders listed above.
3. Select **Properties**. The properties dialog appears.



4. Make sure you are on the **Directory** tab, which is selected by default.
5. Press the **Create** button next to the Application Name field.
6. Assign the folder's name in the **Application name** field.
7. Press **OK**.

If you are creating a new directory for your Web site, you must copy the following files into it from *webroot/siteroot/bin*.

- WebService.dll
- Ektron.Cms.BE.Content.dll
- Ektron.Cms.BE.Font.dll
- Ektron.Cms.BE.Library.dll
- Ektron.Cms.BE.License.dll
- Ektron.Cms.BE.Message.dll
- Ektron.Cms.BE.Module.dll

- Ektron.Cms.BE.Site.dll
- Ektron.Cms.BE.ToolBar.dll
- Ektron.Cms.BE.User.dll
- Ektron.Cms.BE.UserMgmt.dll
- Ektron.Cms.Common.dll
- Ektron.Cms.DataAccess.dll
- Ektron.Cms.DataRW.dll
- Ektron.Cms.Global.dll
- Ektron.Cms.UI.CommonUI.dll
- Ektron.Cms.UI.dll
- Ektron.Cms.WebserviceUI.dll

Providing Web Service Information in Your CMS

CMS400.NET Methods

CMS400.NET implements a single Web Service, ContentWS.asmx, that exposes several methods for retrieving CMS400.NET content. These methods mirror those in the Webservice.dll file.

The methods determine the *kind* of content that is made available (for example, a content block, a collection of content blocks, a summary list) along with the *presentation* of that information (for example, an HTML string or an XML data structure representing content block information).

Below is a list of methods that ContentWS exposes with a brief description of functionality and differences from the applicationAPI implementation.

All methods are demonstrated on the sample page *webroot/siteroot/Workarea/web services/ContentWS.asmx*. For more information, see ["The CMS400.NET Content Sample Page"](#) on page 862.

Method	Exposes this applicationApi method as a web service method	Returns	Changes to standard parameter values
ContentBlock	ecmContentBlock	The content block as an HTML string.	

ContentBlockEx	ecmContentBlockEx	The content block as an XML string. If necessary, performs XSLT transformation on content prior to its return.	
GetContentBlock	ecmContentBlock	An XML data structure representing information in the ContentBlock.	
Collection	ecmCollection	A Collection presented as an HTML string whose format is determined by the display function that you specify. You can create a custom display function in a separate .NET dll.	
GetCollection	ecmCollection	An XML data structure representing information about a Collection.	additional parameter: GetHTML. Possible values: 1 -returns html content for each collection item 0 - does not return it
ListSummary	ecmListSummary	List Summary information as an HTML string.	
GetListSummary	ecmListSummary	An XML data structure representing information about the List Summary.	parameters ShowSummary, StyleInfo, ShowInfo not available.
SingleSummary	ecmSingleSummary	Single Summary information as an HTML string.	
GetSingleSummary	ecmSingleSummary	An XML data structure representing the Single Summary information.	parameters StyleInfo, ShowSummary, ShowInfo not available.
MetaData	ecmMetaData	Meta Data information as an HTML string	does not have the unused spare parameter.

GetMetaData	ecmMetaData	An XML data structure representing Meta Data information	does not have the unused spare parameter.
SearchDisplay	ecmSearchDisplay See Also: " The Search Display Method Parameters " on page 865	Search Display information as an HTML string	
GetSearchDisplay	ecmSearchDisplay See Also: " The Search Display Method Parameters " on page 865	An XML data structure representing the Search Display information	StyleInfo and ShowDate parameters not available
GenerateHTMLMenu	ecmGenerateMenu	Menu details as an html string	Parameters as menu ids (comma separated value)
DropHTMLMenu	ecmDropMenu	Required menu as an html string	Parameters: <ul style="list-style-type: none"> • Root menu id (that is, the generateHTMLMenu value) • MenuId - required menuid • title - required title
GetFormBlock	ecmFormBlock	Form information as an HTML string	Parameters: <ul style="list-style-type: none"> • formId • formdata generated through webserviceUI • formtag required flag (set to true to embed <code><form></code> tag; otherwise, set to false)

GetEventsByCalendar	ecmEvtCalendar	Calendar information as an HTML string	<ul style="list-style-type: none"> • Calendar id • event id • month • year • url • querystring
Show RandomSummary	ecmShowRandomSummary	Randomly displays the summary content of a collection item.	<ul style="list-style-type: none"> • collection ID
ShowRandomContent	ecmShowRandomContent	Randomly displays the content of a content block in a specified collection	<ul style="list-style-type: none"> • collection ID • xsItID
AddContent	ecmAddContent	Lets you programmatically add a new content block to Ektron CMS400 without logging in.	<ul style="list-style-type: none"> • Username • password • Domain • content_title • content_comment • ContentHTML • SummaryHTML • ContentLanguage • FolderID • GoLive • EndDate • MetaInfo • ErrString

The CMS400.NET Content Sample Page

The CMS400.NET content sample page (*webroot\siteroot\Workarea\webservices\ContentWS.asmx*) lists the methods described in the table above. You can access it by signing on to the developer sample site <http://localhost/siteroot/developer/default.aspx> and clicking Web Services from the bottom of the left frame (illustrated below).



ektron CMS400.net

Welcome to the Ektron CMS400.NET Developer Secti

Ektron has compiled a list of several different examples of developi site with Ektron CMS400.NET.

The individual examples, which can be accessed by clicking on the left navigation menu, show how easy it is to create robust Web pa "out-of-the-box" functionality including server controls. Other exa how easy it is to programatically produce Web applications by build server controls.

Click each method on the page to see sample calls and output using various protocols.

When you open the detail page, you see **Test** followed by parameters.

ecmGetContentBlock

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
ContentId:	<input type="text"/>
UserID:	<input type="text"/>
SitePath:	<input type="text"/>
Preview:	<input type="text"/>
SiteLanguage:	<input type="text"/>

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /CMS400Example/Workarea/Webservices/ContentWS.asmx HTTP/1.1
Host: localhost
```

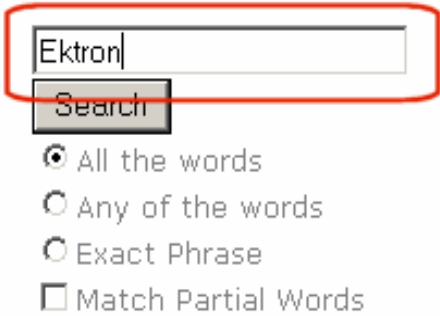
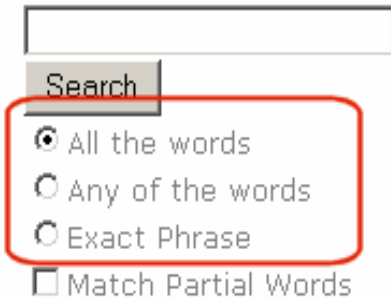
The top section of the page lets you test the method by using the HTTP Get protocol to retrieve data from the sample database. To do this, you must insert valid parameter values.

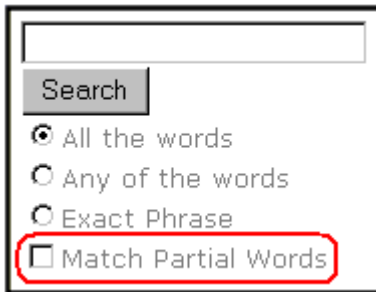
The next section of the CMS400.NET content sample page displays the following kinds of sample output.

- A sample SOAP request and response.
- A sample HTTP GET request and response.
- A sample HTTP POST request and response.

Review the test pages to learn the kind of content you can retrieve from this method.

The Search Display Method Parameters

Parameter	Where Defined in Application API.xxx	How Implemented by CMS
Search Text	n/a	<p>Search text used in search method.</p> 
Search Type	n/a	<p>Type of search used in search method.</p>  <p>Options and - all the words or - any of the words exact phrase - exact phrase</p> <p><u>Note: This parameter is handled automatically by CMS when displayed on Web page. For Web Services, however, this needs to be defined.</u></p>
Recursive	Search	<p>Defines whether search is recursive .</p> <p>1 - recursive search 0 - non-recursive search</p>

Parameter	Where Defined in Application API.xxx	How Implemented by CMS
Starting Folder	Search	<p>Describes path to folder in which search begins. • Here are some examples.</p> <ul style="list-style-type: none"> • \ - root content folder • \Products\RC Cars - begins search in the RC Cars folder.
Allow Fragments	n/a	<p>Selected by user on page that performs search.</p>  <p>Options</p> <p>0 (false) - do not allow fragments.</p> <p>1 (true) - allow fragments.</p> <p>If fragments are allowed, and the user enters "AT" as the search string, the search returns topics with these words: pat, ate, hatter.</p> <p>If fragments are <i>not</i> allowed, and the user enters "AT" as the search string, the search only returns topics with the word: at.</p> <hr/> <p>Note: This parameter is handled automatically by CMS when displayed on Web page. For Web Services, this needs to be defined.</p> <hr/>
Max Number	Search Display	<p>Specify the maximum number of topics to display when a search is performed. For example:</p> <p>0 - unlimited</p> <p>1 - one result</p> <p>7 - seven results</p>

Parameter	Where Defined in Application API.xxx	How Implemented by CMS
Style Info	Search Display	Specify an inline style that is applied to the hyperlinked portion of topics that result from a search. For example: "font-family:arial; font-weight:bold; backgroundcolor:#cccccc; border:solid blue 1pt; margin:2px; width:50%;"
Show Date	Search Display	Specify whether the last modified date is displayed with search results. 0 - date is not displayed. 1 - date is displayed.

Sample Output from One Method

The following is an example of the content that can be retrieved from one of the methods, `GetContentBlock`. As you can see, the HTML content is provided within the `<ContentHtml>` XML element.

GetContentBlock

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentBlockResult xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:ektron:webservices:content:cms400:v45">
  <ErrorMessage />
  <Item>
    <ContentTitle>Plastic Molder #123</ContentTitle>
    <ContentHtml>
      <?xml version="1.0"?> <jobposting> <job-title>Plastics Molder</job-title> <job-number>123</job-number> <description><p>RC International, a leading supplier of consumer level radio controlled vehicles, is looking for an experienced plastics molder to join our team. The candidate will join our highly skilled team of enthusiastic molders to bring to life our next generation products. RC International is experiencing extremely rapid growth and offers many opportunities for advancement.</p></description> <qualifications> <ul> <li>3 years experience in commercial grade plastic molding technologies.</li> <li>Excellent written and oral communication skills.</li> <li>Ability to work with very little supervision.</li> <li>An AS degree in chemistry.</li> </ul> </qualifications> <offering> <p>First and second shift opportunities exist. RC International offers competitive pay, an extensive benefits package including a 401k plan, profit sharing, and health coverage.</p> </offering> <contact> <contact-name>Human Resources Manager</contact-name> <contact-email>jobs@CMS400site.com</contact-email> <contact-fax>1-555-555-5555</contact-fax> </contact> </jobposting>
    </ContentHtml>
    <Xslt1>C:\Inetpub\wwwroot\CMS400Min\XMLFiles\xslt\rc_jobs_display.xsl</Xslt1>
    <Xslt2 />
    <Xslt3 />
    <Xslt4 />
    <Xslt5 />
    <DefaultXslt>1</DefaultXslt>
  </Item>
</ContentBlockResult>
```

The News.aspx Sample (non-XML)

To help you understand how the samples work, this section explains the news sample .NET page, `news.aspx`, which is installed to `webroot/siteroot/WebServiceDemo/`. Specifically, four key sections of code are explained:

- [Identifying the Code Behind File](#)
- [Setting Up an Instance of Web Services](#)
- [Retrieving the Web Service's URL](#)
- [Getting Content Blocks](#)

Identifying the Code Behind File

Here is the first line in the `news.aspx` file:

```
<%@ Page CodeBehind="news.aspx.vb" Language="vb" AutoEventWireup="false"
Inherits="WebServicesDemo.news" %>
```

This line indicates that the page's functionality is set up in the Code Behind Visual Basic file, `news.aspx.vb`. The rest of the `news.aspx` file contains only page display information.

Setting Up an Instance of Web Services

The following code in `news.aspx.vb` sets up an instance of the Web Services. Every page that uses Web Services must contain such a declaration.

```
Dim wsCms400WebService As New ContentWS.ContentWS
```

Getting Content Blocks

The following lines of `news.aspx.vb` code retrieve all content blocks in the sample database's news folder formatted as an HTML string.

```
If (iId > 0) Then
    strHTML = wsCms400WebService.ContentBlock(iId)
    strHTML = Replace(strHTML, ".asp", ".aspx")
    NewsInfo.InnerHtml = strHTML
Else
    strHTML = wsCms400WebService.ListSummary("\Marketing\News", 0, 1, "", "Title,asc",
    "DateModified", 0, "", "")
    strHTML = Replace(strHTML, "news.asp", "WebServiceDemo/news.aspx")
    NewsInfo.InnerHtml = strHTML
End If
```

NOTE Since the database was created with .NET, you could use this code to get the HTML for the requested ID:

```
NewsInfo.InnerHtml = wsCms400WebService.ContentBlock(iId)
```

As you can see, we are assigning to the variable `NewsInfo.InnerHtml` the Web service (`wsCms400WebService`) followed by the `ContentBlock` method. (For a list of methods, see ["CMS400.NET Methods" on page 859](#).)

If the user selects a content block on the List Summary page (that is, clicks the blue text of the content title), that action passes an id value. The blue text is circled in the illustration below.

RC International Ships RC Sportster (8/27/2002 2:46:12 PM)
 RC International, an emerging leader in the RC racing industry, has begun shipments Sportster. The RC Sportster is designed to compete with the entry and mid level RC

A new page appears, using the ContentBlock method to display the selected block.

```
If (iId > 0) Then
  strHTML = wsCms400WebService.ContentBlock(iId)
```

The ContentBlock method retrieves the content block as an HTML string, so the result looks like this.

Click [here](#) for a complete list of operations.

GetContentBlock

Test

To test the operation using the HTTP GET protocol, click the 'Invoke' button.

Parameter	Value
ContentId:	<input type="text"/>
<input type="button" value="Invoke"/>	

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be re

```
POST /CMS400Example/Workarea/Webservices/ContentWS.asmx HTTP/1.1
```

If the user does not select a content block (which is true when the page first appears), the page uses the ListSummary method to display all blocks in the marketing/news folder using the List Summary format.

```
Else
  strHTML = wsCms400WebService.ListSummary("\Marketing\News", 0, 1, "",
"Title,asc", "DateModified", 0, "", "")
```

The ListSummary method retrieves the content blocks as an HTML string, so the page looks like this.

Latest News

[RC International Announces the RC Cheetah \(8/27/2002 2:43:01 PM\)](#)

RC International, a leading RC racing company, has announced the availability of its new Cheetah.

[RC International Hosts RC Conference \(8/27/2002 2:44:44 PM\)](#)

RC International, the fastest growing company in the RC industry, announces plans to h conference in Atlanta, Georgia.

[RC International Ships RC Sportster \(8/27/2002 2:46:12 PM\)](#)

RC International, an emerging leader in the RC racing industry, has begun shipments of i

The News.aspx Sample (XML)

The news.aspx sample file in the /siteroot/WebServiceDemo folder includes all elements described in ["The News.aspx Sample \(non-XML\)" on page 868](#) *plus* demonstrations of how to retrieve XML content.

Setting Up Instances of Web Services

In addition to creating an instance of the Web Service, the XML version of news.aspx.vb creates instances of the ContentBlock and ListSummary methods, which are used on the page.

```
Dim wsCms400WebService As New localhost.CMS400Content()  
Dim gcbCms400ContentXml As localhost.ContentBlockResult  
Dim glsCms400ListSummaryResult As localhost.ListSummaryResult
```

Getting Content Blocks

Look at this code from news.aspx.vb.

```
glsCms400ListSummaryResult = wsCms400WebService.GetListSummary("\Marketing\News",  
0, 1, "", "Title,asc", "DateModified", 0, "", "")  
If (Len(glsCms400ListSummaryResult.ErrorString) = 0) Then  
strHTML = "<table><tr><td><b>Content Block Information Return from the      GetListSummary Web  
service call</b></td></tr>"  
strHTML = strHTML & "<tr><td>&nbsp;</td></tr>"  
For iLoop = 0 To (glsCms400ListSummaryResult.Item.Length - 1)  
strHTML = strHTML & "<tr><td><b>Content ID: </b>" &  
glsCms400ListSummaryResult.Item(iLoop).ContentID & "</td></tr>"  
strHTML = strHTML & "<tr><td><b>Content Title: </b>" &  
glsCms400ListSummaryResult.Item(iLoop).ContentTitle & "</td></tr>"  
strHTML = strHTML & "<tr><td><b>Date Created: </b>" &  
glsCms400ListSummaryResult.Item(iLoop).DisplayDateCreated & "</td></tr>"  
strHTML = strHTML & "<tr><td><b>Date Modified: </b>" &  
glsCms400ListSummaryResult.Item(iLoop).DisplayDateModified & "</td></tr>"  
strHTML = strHTML & "<tr><td><b>Modified By: </b>" &  
glsCms400ListSummaryResult.Item(iLoop).EditorFname & " " &  
glsCms400ListSummaryResult.Item(iLoop).EditorLname & "</td></tr>"  
strHTML = strHTML & "<tr><td><b>Content Teaser: </b></td></tr>"
```

```

        strHTML = strHTML & "<tr><td>" & glsCms400ListSummaryResult.Item(iLoop).ContentTeaser &
"</td></tr>"
strHTML = strHTML & "<tr><td>&nbsp;</td></tr>"
Next

```

This code loops through all content blocks in the Marketing\News folder. For each block, the page uses the Web Service method `glsCms400ListSummaryResult.Item(iLoop)` to display the following XML data.

- Content ID
- Content Title
- Date Created
- DateModified
- Modified By
- Content Teaser

The result looks like this.

Latest News

Content Block Information Return from the GetListSummary Web service call

Content ID: 11

Content Title: RC International Announces the RC Cheetah

Date Created: 25-Apr-2002 04:43:42 PM

Date Modified: 27-Aug-2002 02:43:01 PM

Modified By: Application Administrator

Content Teaser:

RC International, a leading RC racing company, has announced the availability of its new Cheetah.

Content ID: 10

Content Title: RC International Hosts RC Conference

Date Created: 25-Apr-2002 04:42:55 PM

Date Modified: 27-Aug-2002 02:44:44 PM

Modified By: Application Administrator

Content Teaser:

RC International, the fastest growing company in the RC industry, announces plans to host a conference in Atlanta, Georgia.

The lower portion of the page also displays the content block in a ListSummary format, as described in ["Getting Content Blocks" on page 868](#).

Latest News

[RC International Announces the RC Cheetah \(8/27/2002 2:43:01 PM\)](#)

RC International, a leading RC racing company, has announced the availability of its new Cheetah.

[RC International Hosts RC Conference \(8/27/2002 2:44:44 PM\)](#)

RC International, the fastest growing company in the RC industry, announces plans to host a conference in Atlanta, Georgia.

[RC International Ships RC Sportster \(8/27/2002 2:46:12 PM\)](#)

RC International, an emerging leader in the RC racing industry, has begun shipments of its Sportster. The RC Sportster is designed to compete with the entry and mid level RC race

If the user selects a content block and clicks the blue text of any title, a new page appears with

- information about the block (title and any XSLTs used for rendering).
- a demonstration of the GetContentBlock method to display the content block in "raw" HTML or XML format.
- a demonstration of the ContentBlock method to display the content block in processed HTML or XML format.

Here is the code that displays the top section of the page.

```
strHTML = "<table><tr><td><b>Content Block Information from GetContentBlock Web Service: </b></td></tr>"
strHTML = strHTML & "<tr><td><b>Content Block Title: </b>" &
gcbCms400ContentXml.Item.ContentTitle & "</td></tr>"
strHTML = strHTML & "<tr><td><b>Xslt Used for Rendering: </b>"
Select Case (gcbCms400ContentXml.Item.DefaultXslt)
    Case 1
        strTmp = gcbCms400ContentXml.Item.Xslt1
    Case 2
        strTmp = gcbCms400ContentXml.Item.Xslt2
    Case 3
        strTmp = gcbCms400ContentXml.Item.Xslt3
    Case 4
        strTmp = gcbCms400ContentXml.Item.Xslt4
    Case 5
        strTmp = gcbCms400ContentXml.Item.Xslt5
End Select

If (Len(strTmp) = 0) Then
    strTmp = "None Used (HTML Content assumed)"
```

This code uses the gcbCms400ContentXml Web Service to get the content title: gcbCms400ContentXml.Item.ContentTitle.

The code also uses the gcbCms3400ContentXml Web Service to retrieve any XSLTs assigned to the content. If no XSLTs are assigned (as is the case with this sample page), the message "None Used (HTML Content assumed)" appears.

This code results in the following section of the page.

<p style="text-align: center;">Latest News</p> <p>Content Block Information from GetContentBlock Web Service:</p> <p>Content Block Title: RC International Announces the RC Cheetah</p> <p>Xslt Used for Rendering: None Used (HTML Content assumed)</p>
--

Next, the page uses the GetContentBlock method to display the content block in "raw" HTML or XML format. Here is the code that displays that section of the page.

```
strHTML = strHTML & "<tr><td><b>Raw HTML (or XML) from GetContentBlock Web Service: </b></td></tr>"
strHTML = strHTML & "<tr><td>&nbsp;</td></tr>"
strHTML = strHTML & "<tr><td>" & strRawHtml & "</td></tr>"
strHTML = strHTML & "<tr><td>&nbsp;</td></tr>"
```

Finally, the page uses the ContentBlock Web method to display the content block in "processed" format. Here is the code that displays that section of the page.

```
strHTML = strHTML & "<tr><td><b>Processed HTML (or XML) from ContentBlock Web Service: </b></td></tr>"
strHTML = strHTML & "<tr><td>" & strProcessedHtml & "</td></tr>"
strHTML = strHTML & "</table>"
```

Earlier in the news.aspx.vb file, the strProcessedHtml variable was defined as strProcessedHtml = wsCms400WebService.ContentBlock(iId). So, the strProcessedHtml variable uses the wsCms400WebService.ContentBlock method to retrieve content in processed format.

Here is the resulting section of that page. As you can see, the content appears as within a browser.

RC International Announces the RC Cheetah

January 14, 2001 – RC International, a leading RC racing company, has announced the a flagship product, the RC Cheetah. After nearly two years of development the RC Cheetah at the highest level in the RC racing arena. The excitement level of the hard core enthusiasts' demonstrations of the RC Cheetah.

"The RC Cheetah price/performance ration is simply incredible", stated Bob Dickers VP of engineering. "The RC Cheetah has been devastating its competition in all the , and we expect that to continue..."

The major industry magazines are agreeing with Mr. Dickerson's assessment. RC Magazine Cheetah head-to-head with the competition's best and they concluded:

Creating a Page that Provides a Web Service

Below are the typical steps you would follow to create your own page that provides a Web Service.

1. Decide which content blocks you want to provide. Here are some examples:
 - a specific content block (for example, `wsCms400WebService.ContentBlock(2)` would display content block 2 regardless of the URL parameters)
 - all blocks in a folder (the `GetListSummary` method's first parameter is folder)
 - all blocks in a collection (the `Collection` method's first parameter is collection ID)
2. Decide what information you want to provide about those blocks, and how to present it. Review "[CMS400.NET Methods](#)" on page 859 to find the type of content and the kind of information that is available about it.

For example, if you want to display XML data about content (such as content title, date modified, date created), use a method that begins with "Get," such as `GetContentBlock`, `GetCollection`, and `GetListSummary`.
3. Create a new folder within the *siteroot* folder. This folder and some of its subfolders have files that are needed to run Web Services. Place all of your files within the new folder.
4. Using a development tool that supports Web Service integration (for example, Visual Studio 2005), create a .NET (.aspx) file that manages page layout. The file must refer to a Visual Basic or C# file (via a `CodeBehind` command), which manages the page's functionality.
5. Add to each .NET project a Web Reference to the WSDL file. (by default, `webroot/CMS400Min/Workarea/webservices/ContentWS.asmx`). The environment then builds the appropriate proxy class to access the Web

Service's methods. You can use the proxy class within the application to invoke those methods. CMS400.NET sample .aspx pages demonstrate how to do this.

6. Provide to your consumers a URL to the .NET (.aspx) file.

Syndicating Web Content as RSS

This section explains how to add RSS syndication to Ektron CMS400.NET for ASP.NET. RSS syndication allows you to prepare Ektron CMS400.NET content for reading by RSS news feed aggregators.

To make this task easier, Ektron provides sample files to illustrate RSS syndication. You should customize the files to meet your needs.

What is RSS Syndication?

RSS stands for Really Simple Syndication, a common method of syndicating Web content as an XML document. There are several versions of RSS. Ektron has implemented version 2.0.

In Ektron CMS400.NET, you use RSS to prepare a collection list or a list summary for reading by an *aggregator*. An aggregator consumes an RSS feed and displays it to the user. (An aggregator is similar in function to a portal.) There are several popular aggregators, including:

- AmphetaDesk (<http://www.disobey.com/amphetadesk/>)
- NewzCrawler (<http://www.newzcrawler.com/>).

The RSS Specification

The RSS specification is a document: <http://backend.userland.com/rss>. Ektron has implemented the specification, but you can customize the RSS classes to support advanced features.

RSS Syndication in Ektron CMS400.NET

Ektron CMS400.NET allows you to syndicate content by using .NET. Syndicating via .NET requires that you have the Microsoft .NET framework installed on your computer.

With Ektron CMS400.NET, you can syndicate a collection and a list summary via RSS.

Syndicating via .NET

Architecture of RSS Syndication

aspx pages

You can use an aspx page as a URL to dynamically create and return RSS documents. To display an RSS feed on a Web site, the site administrator identifies the URL of the aspx page. The aggregator determines how to use the document's elements, such as <title>, <description> and <link>, to display the syndicated content.

The sample aspx templates (RSS_ListSummary.aspx and RSS_Collection.aspx) are installed by default to *localhost/siteroot* folder. They use the WebService.ContentWS class to retrieve the collection or list summary data. Then, they create CollectionToRSS20 or ListSummaryToRSS20. These classes fill the RSS 2.0 data classes and serialize them to an XML document, which can be returned as an HTTP response.

Since these pages are part of the CMS400.NET sample, you should modify them or create your own aspx pages for your Web site. Place your custom pages in the *yourwebsite/WebServices/* folder so that the web.config file is within the path.

Retrieving a Collection or List Summary as an RSS Feed

The left frame of the corporate.aspx page is hardcoded to retrieve the News Collection. The right frame retrieves a ListSummary.

Creating Your Own aspx Collection Page

To create your own collection page, copy the collection object to an RSS 2.0 object and serialize it. To accomplish this, use these methods.

```
ConvertToRSS(ByVal result CMS400WS.CollectionResult, ByVal  
title As String, ByVal description As String) As RSS20.rss
```

```
Serialize(Optional ByVal stream As System.IO.Stream = Nothing,  
Optional ByVal o As Object = Nothing)
```

Example

```
Dim objRSSConverter As New CollectionToRSS20(Request, Response)  
objRSSConverter.ConvertToRSS(objResult,  
objResult.CollectionTitle, objResult.CollectionDescription)  
objRSSConverter.Serialize()
```

Remarks

The ConvertToRSS method returns the RSS 2.0 object so that you can fill in or modify additional fields as desired before calling Serialize. The Serialize method

has optional parameters that let you send the RSS to a stream other than `HttpResponse`.

You can add code to serialize a different object (such as an RSS 1.0 data class) or to run an XSLT on the serialized XML document before returning it as the HTTP response.

Example

```
Dim objRSSConverter As New CollectionToRSS20(Request, Response)
Dim objRSS20 As rss.RSS20.rss =
objRSSConverter.ConvertToRSS(objResult,
objResult.CollectionTitle, objResult.CollectionDescription)
objRSS20.channel.language = "en-us"
objRSSConverter.Serialize()
```

Creating Your Own aspx List Summary Page

To create your own list summary page, copy a list summary object to an RSS 2.0 object and serialize it. To accomplish this, use these methods.

```
ConvertToRSS(ByVal result As CMS400WS.ListSummaryResult, ByVal
title As String, ByVal description As String) As RSS20.rss
```

```
Serialize(Optional ByVal stream As System.IO.Stream = Nothing,
Optional ByVal o As Object = Nothing)
```

Example

```
Dim objRSSConverter As New ListSummaryToRSS20(Request, Response)
objRSSConverter.ConvertToRSS(objResult, "News", "The latest news")
objRSSConverter.Serialize()
```

Remarks

The `ConvertToRSS` method returns the RSS 2.0 object so that you can fill in or modify additional fields as desired before calling `Serialize`. The `Serialize` method has optional parameters that let you send the RSS to a stream other than `HttpResponse`.

You can add code to serialize a different object (such as an RSS 1.0 data class) or to run an XSLT on the serialized XML document prior to returning it as the HTTP response.

Example

```
Dim objRSSConverter As New ListSummaryToRSS20(Request, Response)
Dim objRSS20 As rss.RSS20.rss =
objRSSConverter.ConvertToRSS(objResult, "Title", "Description")
objRSS20.channel.language = "en-us"
objRSSConverter.Serialize()
```

Learning About Visual Studio

This section introduces some background information about using Microsoft's Visual Studio. For more information, use the help feature installed with Visual Studio 2005 and Microsoft's developer Center (<http://msdn.microsoft.com/vstudio/>).

Grid Layout vs. Flow Layout

NOTE The following definitions are from Visual Studio Help:

Grid Layout - Absolute positioning attributes are inserted into elements that are added, and updated in elements that are moved. Elements can be dragged across the Design view surface. The positioning grid and Snap to Grid are available.

Flow Layout - Elements are added without absolute positioning attributes. Web browsers arrange elements in the order that they occur on the page, from top to bottom. You cannot drag elements across the Design view surface or use the positioning grid.

Grid layout is the default, which means that all controls drawn to the Web form in the designer window will have absolute positioning. Here is an example.

```
<body MS_POSITIONING="GridLayout">
  <form id="Form1" method="post" runat="server">
    <asp:Button id="Button1" style="Z-INDEX: 101; LEFT: 160px; POSITION: absolute; TOP: 80px"
runat="server" Text="Button"></asp:Button>
    <asp:Button id="Button2" style="Z-INDEX: 102; LEFT: 480px; POSITION: absolute; TOP: 88px"
runat="server" Text="Button"></asp:Button>
    <asp:GridView id="GridView1" style="Z-INDEX: 103; LEFT: 208px; POSITION: absolute; TOP:
152px" runat="server"></asp:GridView>
  </form>
</body>
```

In Grid layout, you can position your controls like a WYSIWYG editor with no knowledge of HTML. However, because absolute positioning is not rendered consistently by all browsers, the page layout can be flexible based on the size of other controls on the page, and the Web browser window.

When other controls are dynamically populated, such as a GridView, controls that appear beneath it in the Web form would be obscured if they were positioned absolutely at design time. In addition, when utilizing globalization of pages with different languages, the size of text areas can vary and cause obstructions.

Customizing Validation

Customizing Validation Options

In the `cms_config.aspx` file, you can specify a set of validation options for plain text and calculation field types. (Data in other field types cannot be validated.) For these field types, you can assign standard and custom validation checks. The checks are applied when data is inserted into one of the field types, and when the user tries to save a Data Design document whose fields have validation attributes.

You can modify the standard options and enter your own criteria for each field. You can also establish dependencies between fields. For example, a value is only required for a field if a certain Checkbox field is checked.

Validation Elements in the Configuration Data

The default configuration file includes standard validation options for plain text and calculation field types.

```
<validation name="calculation">
  <choice name="none" treeImg="text">
    <caption localeRef="dlgNV8n" />
    <schema datatype="string" />
  </choice>
</validation>
```

and

```
<validation name="plaintext" visible="false">
  <choice name="none" treeImg="text">
    <caption localeRef="dlgNV8n" />
  </choice>
</validation>
```

The validation sections let you control the drop-down list of validation choices for a field type (for example, Plain Text). Here is an overview of that section of the `cms_config.aspx` file.

```
<datadesign>
  <validation> (0 or more)
    <choice> (0 or more)
      <caption>
      <schema>
      <calculate>
    </choice>
  </validation> OR
  <regex> OR
  <script> OR
  <xpath>
  <validate>
    <regex> OR
    <script> OR
    <xpath>
  </validate>
  <errorMessage> (optional)
</datadesign>
```

Details are provided below.

Validation Attributes

Attribute	Description	Possible values
visible	Controls whether the Validation field appears on the field's Properties dialog.	true, false
enabled	Controls whether the Validation field is active or "grayed out" on the field's Properties dialog. If set to true , the Validation field is active; if false , it appears but is grayed out.	true, false
name	The field type to which the validation configuration data applies. A separate <validation> element must exist for each field type.	plaintext, calculation

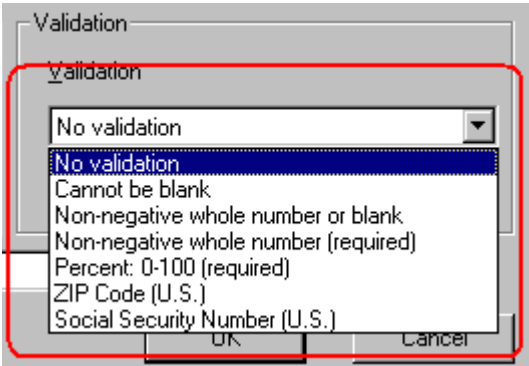
Validation Sub-elements

Element	Description	For more information, see
Choice	The calculation or validation expression for standard fields	"Choice Sub-element" on page 881
Custom	The calculation or validation expression for custom fields	"Defining Custom Validation" on page 886

Choice Sub-element

Every item in the validation drop-down list must be defined within a set of <choice> tags. Within the <choice> tags, you define a caption, a schema, and either a calculation or validation expression.

The default validation choices appear below.



Choice Attributes

Attribute	Description
name	Assign a new name to each choice.
treeImg	<p>The icon to display for this field in the Select Field or Group dialog. See "Icons on the Select Field or Group Screen" on page 891.</p> <p>Unlike toolbar icons, you cannot create your own icons.</p> <p><u>Note: This attribute only applies to the Data Design of a Smart Form -- it does not apply to HTML forms.</u></p>

Choice Sub-Elements

Element	Description	For more information, see
caption	Defines the displayed text for this choice. The attributes and description are the same as <caption> elements for commands.	
schema	<p>Defines a W3C XML schema (WXS) definition for this choice. The definition may be a simple type defined by the datatype attribute and/or a WXS fragment.</p> <p><u>Note: This attribute only applies to the Data Design of a Smart Form -- it does not apply to HTML forms.</u></p>	"Defining a Schema Fragment" on page 883
calculate	Defines an expression which normalizes a value prior to checking validation	"Defining a Calculation" on page 884
validate	Defines an expression that determines if a value is valid	"Defining Validation" on page 885

Defining a Schema Fragment

When defining a schema fragment, the "xs" namespace prefix is required for WXS tags. Also, the WXS fragment must be valid for inclusion in an xs:restriction (simple type). That is,

```
<xs:simpleType>
  <xs:restriction>
    ...fragment...
  </xs:restriction>
</xs:simpleType>
```

Simple Data Types

The simple data types are defined by W3C XML Schema definition language 1.0.

NOTE The datatype value should *not* include a namespace prefix. For example datatype="string" is correct; datatype="xs:string" is incorrect

Examples

- Using only a datatype attribute

```
<schema datatype="string" />
```

- Using only a schema fragment

```
<schema>
  <xs:simpleType>
    <xs:union memberTypes="xs:nonNegativeInteger">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="0" />
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
</schema>
```

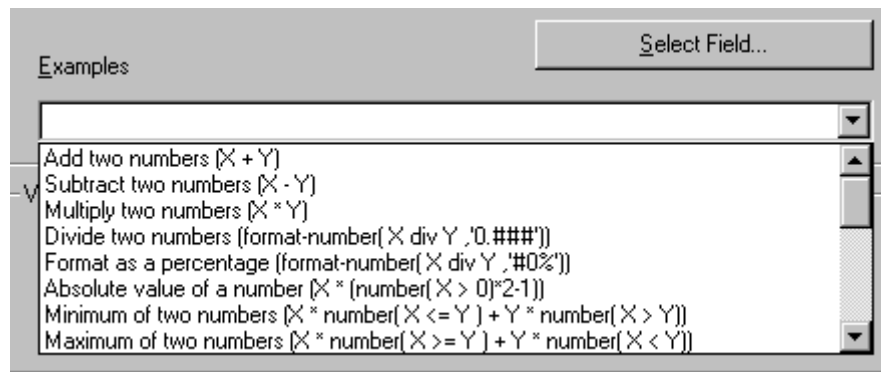
- Using a datatype attribute *and* a schema fragment

```
<schema datatype="string">
  <xs:minLength value="1" />
</schema>
```

Defining a Calculation

Use the `<calculate>` element to define an expression that normalizes a value prior to checking validation. For example, a calculation can truncate digits in a decimal, remove excess white space, or capitalize text. The expression must return a value of the same data type and format as the original value.

Several sample calculations are delivered with **Ektron CMS400.NET**. They appear when the user clicks the **Examples** drop-down on the Calculated Field dialog. These calculations are explained in the “Explanation of Examples” section of the “Using the Data Designer” chapter of the Administrator Manual.



You can only define one expression for `<calculate>` and `<validate>`.

The expression element choices are listed below. Use whichever language is easiest.

- [Regular Expression \(JScript\)](#)
- [JavaScript](#)
- [XPATH](#)

Expression	Description
Regular Expressions (JScript)	
<code><regexp></code>	Defines a regular expression supported by JScript. You can define a regular expression <i>either</i> between the tags <i>or</i> using attributes. If between the tags, the expression must begin with a slash (/) character. The g, i, and m flags are allowed. For example: <code><regexp>/\S+</regexp></code> .
<code><regexp patternings-exp-pattern></code>	A regular expression pattern (without the "/" chars).
<code><regexp global="true false"></code>	Specifies whether the pattern matches only the first occurrence or all occurrences within the text. This corresponds to the g flag.
<code><regexp ignorecase="true false"></code>	Specifies if the match is case-sensitive. This corresponds to the "i" flag.
<code><regexp multiline="true false"></code>	Specifies if the match, when using ^ and \$, is applied to each line in text that has multiple lines. This corresponds to the "m" flag.
<code><regexp wholeline="true false"></code>	Specifies whether the pattern applies to the whole text or not. This is the same as placing "^(" at the beginning of the pattern, and ")\$" at the end of the pattern.
JavaScript	
<code><script value="javascript-expression"></code>	Specifies a JavaScript expression to be evaluated. The field's value is available in a property named 'this.text'. For example <code><script value="this.text.toUpperCase()" /></code>
XPATH (See Also: "Learn More about XPath" on page 893)	
<code><xpath select="xpath-expression"></code>	Specifies an XPath expression to be evaluated. The field's value is available using ".".

Defining Validation

Use the `<validate>` element to define an expression that determines if a value is valid. The expression must return a Boolean (true or false) result. For example:

```
<validate>
  <regex>/^\d+$/</regex>
</validate>
```

To construct the <validate> element, use the same expression element options as <calculate>. See Also: "Defining a Calculation" on page 884

Defining an Error Message

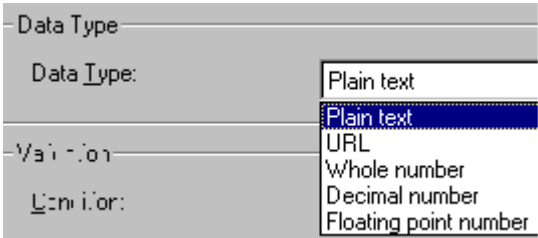
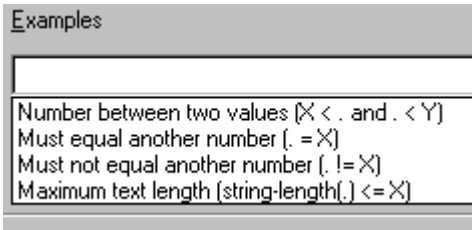
Use the <errormessage localeRef="id"> element to define a message to display when the data is not valid. For example, "Must be a number between 1 and 10, inclusive."

The text may be within the <errormessage> tags or referenced using localeRef into the localeNNNN.xml file.

Defining Custom Validation

NOTE This section only applies to the Data Design of a Smart Form -- it does not apply to HTML forms.

Use the <custom> element to change the standard values that appear in the following fields of the Custom Validation dialog.

Field	Screen image
Data Type	
Examples	

To change the list of options, modify the <custom> element of the cms_config.aspx file. This section describes the <custom> element's attributes and child elements.

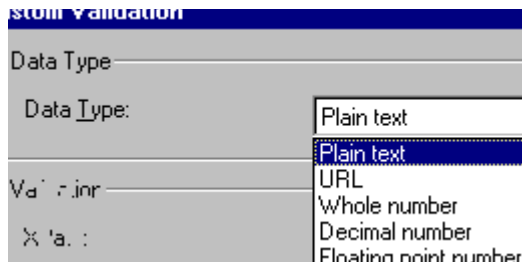
```
<datadesign>
  <validation>
```

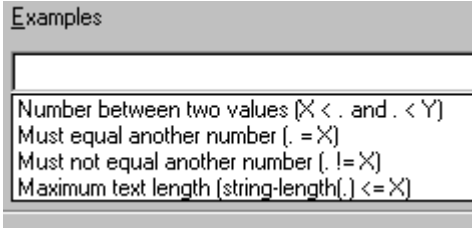
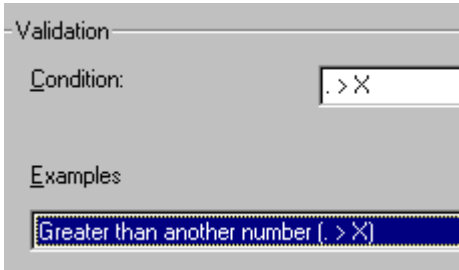
```

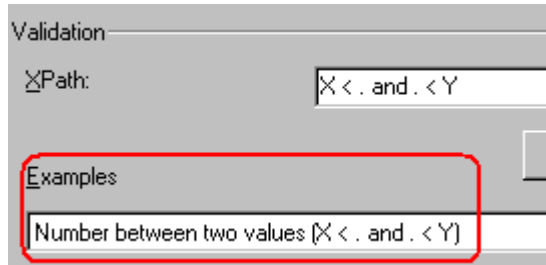
:
<custom> (optional tag)
  <caption>
  <selections name="datatype">
    <listchoice> (0 or more)
  <selections name="examples">
    <listchoice> (0 or more)

```

Element or Attribute	Description
<custom>	Specifies basic data types available when customizing validation. The types are defined in the <listchoice> elements (see below).
<custom visible="true false">	Controls whether the Custom Validation field appears on the Properties dialog.
<custom enabled="true false">	Controls whether the Custom Validation field is active or "grayed out" on the Properties dialog. If set to true , the Custom Validation field is active; if false , it is grayed out.
<caption localeRef="id">	Specifies text to display in the validation drop-down list. The default caption is "(Custom)" .
Selections element for Data Type field	
<selections name="datatype">	The name must be datatype.
<selections enabled="true false">	Controls whether the drop-down list is active or "grayed out" on the Properties dialog.
<selections visible="true false">	Controls whether the drop-down list appears on the Properties dialog.
Listchoice element for Data Type field	
<listchoice>	Defines the values in the Custom Validation dialog Data Type drop-down list.

Element or Attribute	Description
<code><listchoice value="simple-data-type"</code>	<p>Enter each data type that should appear in the Custom Validation dialog's Data Type drop-down list.</p>  <p>The simple data types are defined by W3C XML Schema definition language 1.0.</p> <p>See Also: "Defining a Schema Fragment" on page 883</p>
<code><listchoice treelmg="id"</code>	<p>Specifies the icon to display for this field in the Select Field or Group dialog. See "Icons on the Select Field or Group Screen" on page 891.</p> <p>Unlike toolbar icons, you cannot create your own.</p>
<code><listchoice localeRef="id"</code>	<p>The text that describes this Data Type on the Custom Validation dialog.</p> <p>This element can refer to a string in the localeNNNN.xml file. Or, you can enter the string between the <code><listchoice></code> tags.</p>
<code><listchoice default="true false"</code>	<p>Use this attribute to indicate the default choice in the Custom Validation dialog's Data Type drop-down list.</p>
Selections element for Examples field	
<code><selections name="examples"></code>	<p>The name must be "examples".</p>

Element or Attribute	Description
<selections enabled="true false">	<p>Determines if the Examples drop-down list and label are active or "grayed out" in the Custom Validation dialog.</p> 
<selections visible="true false">	<p>Determines if the Examples drop-down list and label appear.</p>
Listchoice Element for Examples field	
<listchoice>	<p>Defines the values in the Examples drop-down list.</p>
<listchoice value="xpath-expression-example">	<p>The XPath expression appears in the Examples drop-down list of the Custom Validation dialog.</p> 

Element or Attribute	Description
<listchoice localeRef="id" -	<p>The text that describes the examples on the Custom Validation dialog.</p>  <p>This element can refer to a string in the localeNNNN.xml file. Or, you can enter the string between the <listchoice> tags.</p>

Saving Invalid Documents

Use the `publishinvalid` attribute of the `<standard>` element to determine if an invalid document should be saved. This attribute lets you decide if a user can save an XML document when the data in one or more fields does not satisfy the validation criteria.

In Data Entry mode, an example would be if a field requires a non-negative whole number, but the user does not insert a value in that field.

If the `publishinvalid` attribute's value is `true`, content is not checked for validity when it is saved.

If `false`, the content is checked for validity during a save. If the content is valid, it is saved. If it is invalid, the user is notified, and a custom script can be created to allow the content to be saved or prevent it.

NOTE If the `publishinvalid` attribute is not defined in the `<standard>` element, the default is `true` so that Ektron CMS400.NET is backwards compatible with previous releases.

Custom Script that Handles Saving Invalid Files

To determine whether or not an invalid XML document should be saved, write a client-side script that is called when invalid content is found. The routine should ask if the user wants to save the invalid content. The script should return `True` to save the content, or `False` to abort the save.

Below is an example of such a script.

```
<script language="JavaScript1.2">
<!--
eWebEditPro.instances["MyEditor1"].onerror = myOnErrorHandler;

function myOnErrorHandler()
```



```

{
  if (EWEP_STATUS_INVALID == this.status && "save" == this.event.source)
  {
    var strMsg = "Content is invalid.";
    strMsg += "\nError Code: " + this.event.reason;
    strMsg += "\nError Reason: " + this.event.message;
    alert(strMsg);
    return false; // prevent save
  }
}
//-->
</script>

```

The onerror Event

If the content is invalid, an onerror event is generated. This event provides two additional properties for the event object when it fires:

- reason (a numeric error code)
- message (text describing the error)

As with the regular onerror event, the `source` property is available. You can display the values of these properties in an error message that informs the user why the document is not valid.

Calculated Fields

A Calculated Field lets you include values that are calculated, typically from values in other fields. For example, you can multiply two field values. The equation used is an XPath expression.

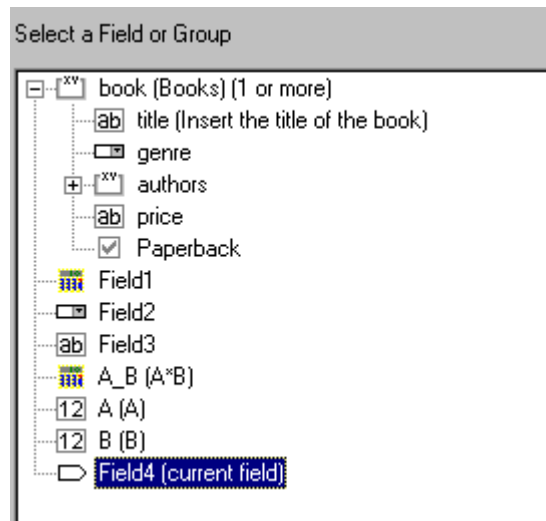
You can validate a Calculated Field, just like the Plain Text Field. For instance, the total of a series of numbers should be less than 100.










For more information about calculated fields, see the **Ektron CMS400.NET Administrator Manual** chapter "Using the Data Designer."




Icons on the Select Field or Group Screen

NOTE [This section only applies to the Data Design of a Smart Form -- it does not apply to HTML forms.](#)

Use the following icons to denote a field on the Select A Field or Group Screen (illustrated below). These are the valid values for the `treeImg` attribute.



Icon	Name
	calculation
	calendar
	checkbox
	droplist
	fieldset
	hidden
	hyperlink
	number
	password
	picture

Icon	Name
	richarea
	text
	textbox

Learn More about XPath

Xpath Operators

Content copied from www.w3schools.com/xpath.

Operator	Description	Example	Result
Numerical expressions perform arithmetic operations on numbers. XPath converts each operand to a number before performing an arithmetic expression.			
+	addition	6 + 4	10
-	subtraction	6 - 4	2
*	multiplication	6 * 4	24
div	division	8 div 4	2
mod	division remainder	5 mod 2	1
Equality/ greater/ less than expressions test equality between two values			
=	equals	price = 9.80	true (if price is 9.80)
!=	is not equal	price! = 9.80	false
<	less than	price < 9.80	false (if price is 9.80)
< =	less than or equal to	price <= 9.80	true

Operator	Description	Example	Result
>	greater than	price > 9.80	false
>=	greater than or equal to	price >= 9.80	true
Boolean expressions compare two values			
or	or	price = 9.80 or price = 9.70	true (if price is 9.80)
and	and	price <=9.80 and price = 9.70	false
Identifying a path and element			
.	the current element	. > 100	true if element exceeds 100
..	the current element's parent	count(../*)	counts the number of elements at the same level as the current element
Grouping and separating			
[]	predicate	../*[0]	../*[0] - returns the parent's first child element
	specify multiple elements	sum(X Y Z)	If X=1 and Y =2 and Z=3, sum(X Y Z) = 6

Xpath Functions

Content copied from www.w3schools.com/xpath.

Operator	Description	Syntax	Example
last	Returns the position number of the last node in the processed node list	number=last()	
position	Returns the position in the node list of the node that is currently being processed	number=position()	

Operator	Description	Syntax	Example
count	Returns the number of nodes in a node-set	number=count(node-set)	
name	Returns the name of a node	string=name(node)	
string	Converts the value argument to a string	string(value)	string(314) Result: '314'
concat	Returns the concatenation of all its arguments	string=concat(val1, val2, ..)	concat('The', ' ', 'XML') Result: 'The XML'
starts-with	Returns true if the first string starts with the second string. Otherwise, it returns false.	bool=starts-with(string, substr)	starts-with('XML', 'X') Result: true
contains	Returns true if the second string is contained within the first string. Otherwise, it returns false.	bool=contains(val, substr)	contains('XML', 'X') Result: true
substring-after	Returns the part of the string in the string argument that occurs after the substring in the substr argument	string=substring-after(string, substr)	substring-after('12/10', '/') Result: '10'
substring-before	Returns the part of the string in the string argument that occurs before the substring in the substr argument	string=substring-before(string, substr)	substring-before('12/10', '/') Result: '12'
substring	Returns a part of the string in the string argument	string=substring(string, start, length)	substring('Beatles', 1, 4) Result: 'Beat'
string-length	Returns the number of characters in a string	number=string-length(string)	string-length('Beatles') Result: 7
normalize-space	Returns the whitespace-normalized version of a passed string. All leading and trailing whitespace is stripped, and all sequences of whitespace get combined to one single space.	normalize-space('string')	normalize-space(' some text ') would return some text

Operator	Description	Syntax	Example
translate	normalize	string=translate(value,string1,string2)	<ul style="list-style-type: none"> translate('12:30','30','45') Result: '12:45' translate('12:30','03','54') Result: '12:45' translate('12:30','0123','abcd') Result: 'bc:da'
boolean	Converts the value argument to Boolean and returns true or false	bool=boolean(value)	
not	Returns true if the condition argument is false, and false if the condition argument is true	bool=not(condition)	not(false())
true	Returns true	true()	number(true()) Result: 1
false	Returns false	false()	number(false()) Result: 0
lang	Returns true if the language argument matches the language of the xsl:lang element. Otherwise, it returns false.	bool=lang(language)	
number	Converts the value argument to a number	number=number(value)	number('100') Result: 100
sum	Returns the total value of a set of numeric values in a node-set	number=sum(nodeset)	sum(/cd/price)
floor	Returns the largest integer that is not greater than the number argument	number=floor(number)	floor(3.14) Result: 3
ceiling	Returns the smallest integer that is not less than the number argument	number=ceiling(number)	ceiling(3.14) Result: 4

Operator	Description	Syntax	Example
round	Rounds the number argument to the nearest integer	integer=round(number)	round(3.14) Result: 3

XPath References

To learn more about XPath, check these Web pages.

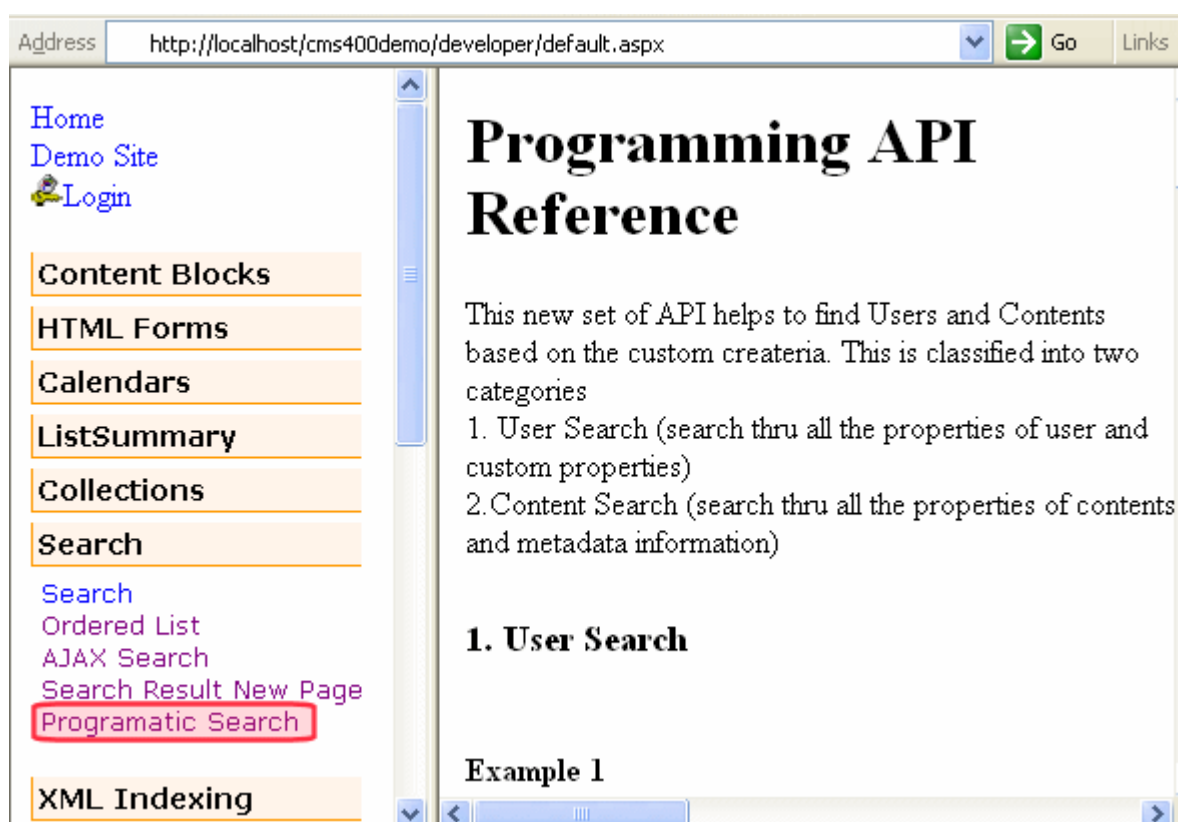
Topic	URL
W3C Spec - XPath 1.0 spec	http://www.w3.org/TR/xpath
Tutorial	www.w3schools.com/xpath/default.asp
Quick reference card	www.mulberrytech.com/quickref/XSLTquickref.pdf

Programmatic Search API

NOTE The Programmatic Search API does not support recursive searches. However, using code behind you can use the Search server control to programmatically search recursively. See Also: "Search Server Control" on page 318

Ektron CMS400.NET provides an API that lets you create a programmatic search on users and content. For example, you could use it to find all membership users whose zip code is 03031.

To see an example of the search API, go to the developer's page (<http://yourWebserver/cms400developer/default.aspx>) and click **Search > Programmatic Search**.



Note that below each example on the page is **run this example**. Click this link to run the search and view the results.

Guidelines for Creating a Search

To learn how to use the Search API, navigate to <http://your Web server/cms400developer/default.aspx> in your browser. Next, click **Search > Programmatic Search** and then study the examples. Note that there are two separate examples. One example is for searching Users and the other for searching Content. These examples are provided in both Visual Basic and C#.

Here are guidelines to follow when creating a search.

IMPORTANT!

Use these guidelines in conjunction with the web page shown above. The code below is from the User Search example.

1. A search consists of one or more conditions. Each condition consists of an operator (and, OR, Like), a value, and the field whose value you are setting. See illustration below.

```
Dim isMemberShip As UserSearchCondition = New UserSearchCondition '''Condition
isMemberShip.setType = SearchType.Equal '''Operator
isMemberShip.setValue = 1 '''Value. The value can be integer,string,date and boolean should be
match with db type
isMemberShip.setVariable = Users.membership_user '''Field
Dim isInEktron As UserSearchCondition = New UserSearchCondition '''Condition
isInEktron.setType = SearchType.Equal '''Operator
isInEktron.setValue = "03031" '''Value. The value can be integer,string,date and boolean
should be match with dynamic_data_tbl labels type
isInEktron.setVariable = "customproperties.zip" '''Condition
```

WARNING!

If the field is a date, you can only use the following operators: EQUAL, NOT EQUAL, GREATER THAN or LESS THAN.

2. After all conditions are declared, declare the logical relationship between them. In other words, must the search criteria satisfy all or any conditions? In the sample code below, the search only returns users that satisfy both declared conditions. This is indicated by the AND operator.

```
Dim condition As UserSearchCondition = New UserSearchCondition
condition.setType = SearchType.AND
condition.AddCondition(isInEktron)
condition.AddCondition(isMemberShip)
```

3. Execute the search.

```
Dim search As New SearchManager
Dim result As UserData() = search.Execute(condition)
search = Nothing
```

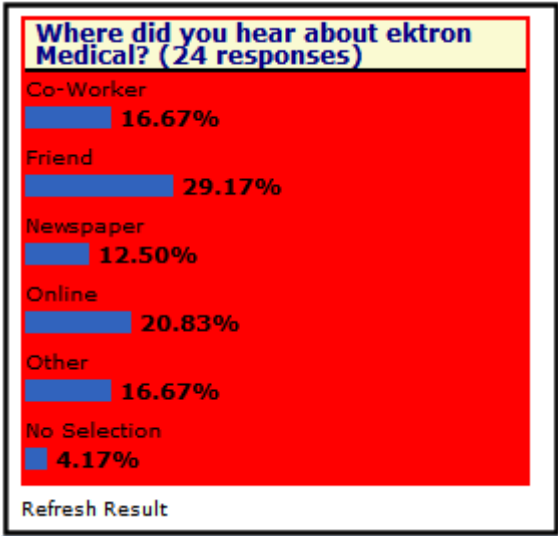
Cascading Style Sheets


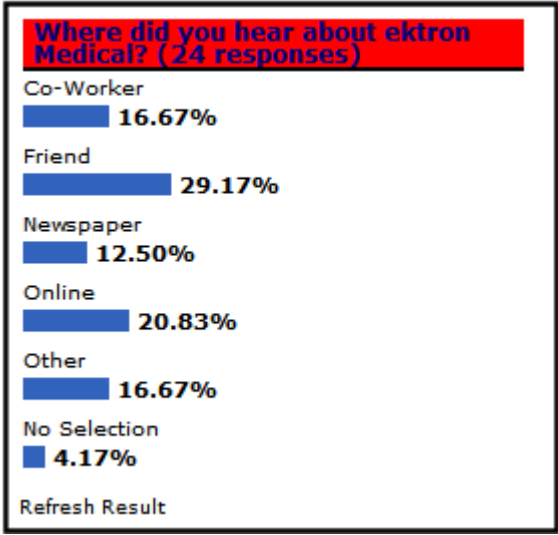
Below are descriptions of the various Cascading Style Sheets used in **Ektron CMS400.NET**.

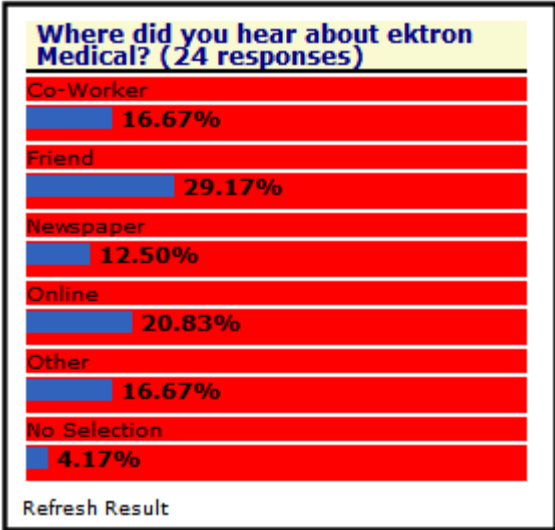
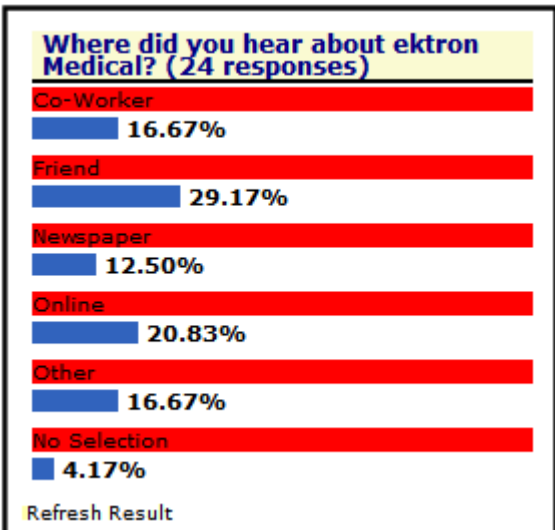
reportchart.css

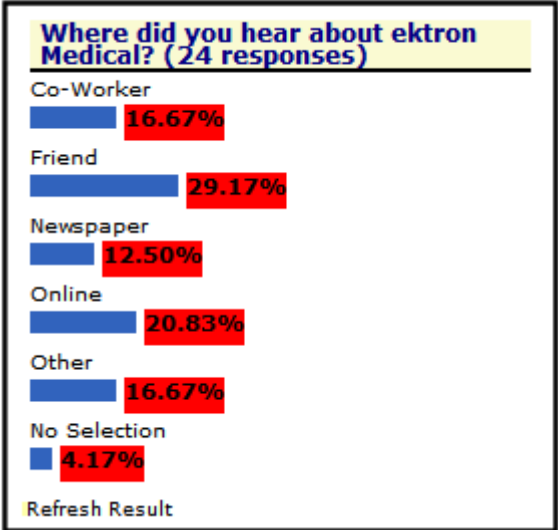

The reportchart.css file allows you to change the look and feel of the postback reports used with forms. This files default location is

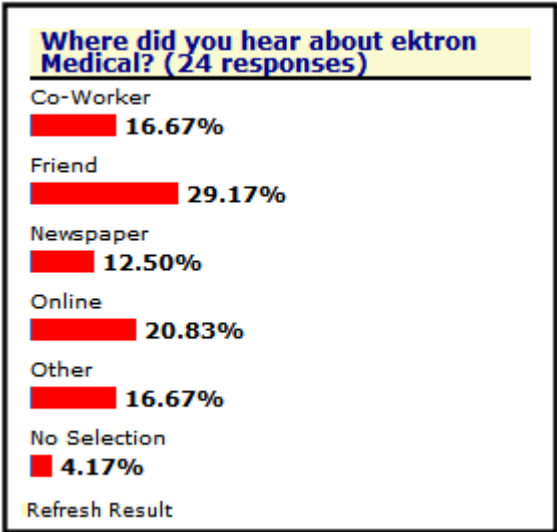
`\webroot\siteroot\Workarea\csslib.`

Class	Controls												
TABLE.tblreport	<p>The area where the report appears. The following example shows the area where the report appears in red.</p>  <table><caption>Where did you hear about ektron Medical? (24 responses)</caption><tbody><tr><td>Co-Worker</td><td>16.67%</td></tr><tr><td>Friend</td><td>29.17%</td></tr><tr><td>Newspaper</td><td>12.50%</td></tr><tr><td>Online</td><td>20.83%</td></tr><tr><td>Other</td><td>16.67%</td></tr><tr><td>No Selection</td><td>4.17%</td></tr></tbody></table> <p>Refresh Result</p>	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Co-Worker	16.67%												
Friend	29.17%												
Newspaper	12.50%												
Online	20.83%												
Other	16.67%												
No Selection	4.17%												

Class	Controls
A.refreshlink	<p>The Refresh Results link that appears below the report. The following example shows the Refresh Results link is highlighted in red.</p> 
A.refreshlink:ACTIVE	The Refresh Results link as you are clicking on it.
A.refreshlink:VISITED	The Refresh Results link after you have clicked on it.
.headreport	<p>The header area where the question and number of responses appear. The following example shows the header area in red.</p> 

Class	Controls														
.tblcol1, .tblcol2	<p>The table columns where the options and results appear. The following example shows the table columns in red.</p>  <p>The screenshot shows a poll interface with a title 'Where did you hear about ektron Medical? (24 responses)'. Below the title are six rows, each representing a response category. Each row has a blue bar indicating the percentage of responses and the percentage value itself. The categories and their percentages are: Co-Worker (16.67%), Friend (29.17%), Newspaper (12.50%), Online (20.83%), Other (16.67%), and No Selection (4.17%). At the bottom of the poll interface is a 'Refresh Result' button.</p> <table border="1"><thead><tr><th>Category</th><th>Percentage</th></tr></thead><tbody><tr><td>Co-Worker</td><td>16.67%</td></tr><tr><td>Friend</td><td>29.17%</td></tr><tr><td>Newspaper</td><td>12.50%</td></tr><tr><td>Online</td><td>20.83%</td></tr><tr><td>Other</td><td>16.67%</td></tr><tr><td>No Selection</td><td>4.17%</td></tr></tbody></table>	Category	Percentage	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Category	Percentage														
Co-Worker	16.67%														
Friend	29.17%														
Newspaper	12.50%														
Online	20.83%														
Other	16.67%														
No Selection	4.17%														
.optiontextcell	<p>The area where the options appear. The following example shows the option area in red.</p>  <p>This screenshot is identical to the one above, showing the same poll results. The categories and percentages are: Co-Worker (16.67%), Friend (29.17%), Newspaper (12.50%), Online (20.83%), Other (16.67%), and No Selection (4.17%). The 'Refresh Result' button is at the bottom.</p> <table border="1"><thead><tr><th>Category</th><th>Percentage</th></tr></thead><tbody><tr><td>Co-Worker</td><td>16.67%</td></tr><tr><td>Friend</td><td>29.17%</td></tr><tr><td>Newspaper</td><td>12.50%</td></tr><tr><td>Online</td><td>20.83%</td></tr><tr><td>Other</td><td>16.67%</td></tr><tr><td>No Selection</td><td>4.17%</td></tr></tbody></table>	Category	Percentage	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Category	Percentage														
Co-Worker	16.67%														
Friend	29.17%														
Newspaper	12.50%														
Online	20.83%														
Other	16.67%														
No Selection	4.17%														

Class	Controls														
.percentcell	<p>The area where the result's percentage information appears. The following example shows the result's percentage information area in red.</p>  <table><thead><tr><th>Source</th><th>Percentage</th></tr></thead><tbody><tr><td>Co-Worker</td><td>16.67%</td></tr><tr><td>Friend</td><td>29.17%</td></tr><tr><td>Newspaper</td><td>12.50%</td></tr><tr><td>Online</td><td>20.83%</td></tr><tr><td>Other</td><td>16.67%</td></tr><tr><td>No Selection</td><td>4.17%</td></tr></tbody></table>	Source	Percentage	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Source	Percentage														
Co-Worker	16.67%														
Friend	29.17%														
Newspaper	12.50%														
Online	20.83%														
Other	16.67%														
No Selection	4.17%														
.barcell	<p>The cell where the bar graph appears. The following example shows the cell area in red.</p>  <table><thead><tr><th>Source</th><th>Percentage</th></tr></thead><tbody><tr><td>Co-Worker</td><td>16.67%</td></tr><tr><td>Friend</td><td>29.17%</td></tr><tr><td>Newspaper</td><td>12.50%</td></tr><tr><td>Online</td><td>20.83%</td></tr><tr><td>Other</td><td>16.67%</td></tr><tr><td>No Selection</td><td>4.17%</td></tr></tbody></table>	Source	Percentage	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Source	Percentage														
Co-Worker	16.67%														
Friend	29.17%														
Newspaper	12.50%														
Online	20.83%														
Other	16.67%														
No Selection	4.17%														

Class	Controls														
.resultbar	<p>The result's bar graph bar. The following example shows the cell area in red.</p>  <table border="1"> <caption>Where did you hear about ektron Medical? (24 responses)</caption> <thead> <tr> <th>Source</th> <th>Percentage</th> </tr> </thead> <tbody> <tr> <td>Co-Worker</td> <td>16.67%</td> </tr> <tr> <td>Friend</td> <td>29.17%</td> </tr> <tr> <td>Newspaper</td> <td>12.50%</td> </tr> <tr> <td>Online</td> <td>20.83%</td> </tr> <tr> <td>Other</td> <td>16.67%</td> </tr> <tr> <td>No Selection</td> <td>4.17%</td> </tr> </tbody> </table> <p>Refresh Result</p>	Source	Percentage	Co-Worker	16.67%	Friend	29.17%	Newspaper	12.50%	Online	20.83%	Other	16.67%	No Selection	4.17%
Source	Percentage														
Co-Worker	16.67%														
Friend	29.17%														
Newspaper	12.50%														
Online	20.83%														
Other	16.67%														
No Selection	4.17%														

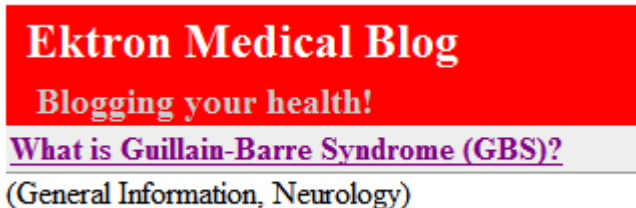
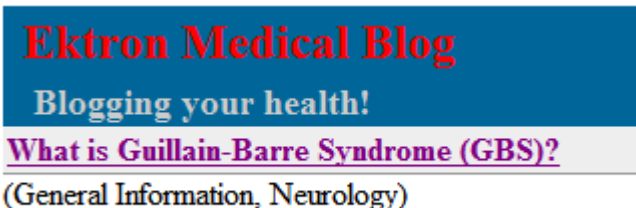
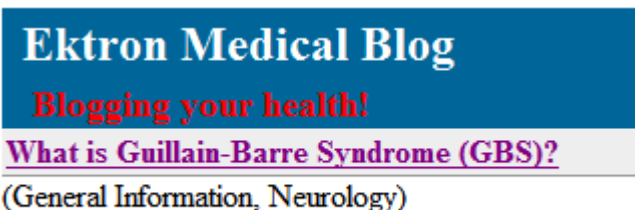
blogs.css

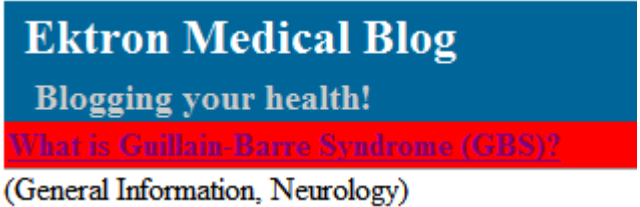

This file is located in `webroot/siteroot/Workarea/csslib`. It is used to customize the look of the Blog server controls.




The table below contains the following sections:




- "Layout Styles" on page 904
- "Header Styles" on page 905
- "Entry Styles" on page 906
- "Calendar Styles" on page 906
- "Blogroll Styles" on page 909
- "Blog Categories Styles" on page 910
- "Blog Recent Posts Styles" on page 911




Style Class	Description
Layout Styles	


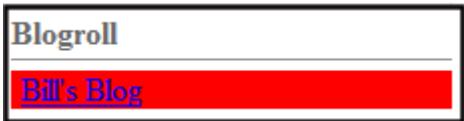
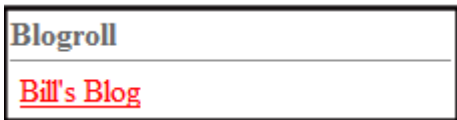
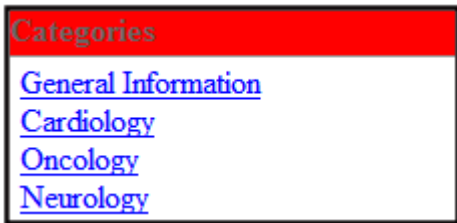
Style Class	Description
div.blog	Affects the blog entry information contained in the Blog server control
div.blogEntries	The border to the right of the blog entries in the Blog server control
div.blogSidebar	The area where the blog calendar, blog roll, categories and recent posts are located in the Blog server control.
Header Styles	
div.blogHeader	<p>The header area for the blog. Affects the Blog and BlogEntries server controls. The following example shows the header area for the blog in red.</p> 
h1.blogTitle	<p>The title of the blog. Affects the Blog and BlogEntries server controls. The following example shows the blog title in red.</p> 
h2.blogSubhead	<p>The subhead (Tagline) of the blog. Affects the Blog and BlogEntries server controls. The following example shows the blog Subhead in red.</p> 


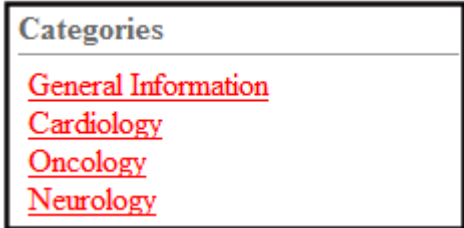

Style Class	Description
span.date	The date in the blog title. Affects the Blog and BlogEntries server controls.
Entry Styles	
div.entry	The border line that separates blog entries. Affects the Blog and BlogEntries server controls.
h3.entryTitle	<p>The title area for blog entries. Affects the Blog and BlogEntries server controls. The following example shows the title area for blog entries in red.</p> 
div.entryBody	The blog entry body. Affects the Blog and BlogEntries server controls.
div.entryFooter	<p>The blog entry footer information. Affects the Blog and BlogEntries server controls. The following example shows the footer information for blog entries in red.</p> 
Calendar Styles	


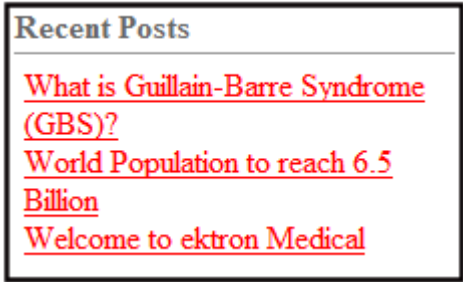
Style Class	Description
table.calendarTable	<p>The table area containing the calendar. Affects the Blog and BlogCalendar server controls. The following example shows the table area in red.</p> 
tr.monthRow	<p>The row that contains month information in the calendar. Affects the Blog and BlogCalendar server controls. The following example shows the month row in red.</p> 
td.prevMonth, td.nextMonth	<p>The area of the calendar for the previous and next month. Affects the Blog and BlogCalendar server controls. The following example shows the previous and next month area in red.</p> 

Style Class	Description
td.prevMonth a, td.nextMonth a	<p>The text information for the previous and next month area of the calendar. Affects the Blog and BlogCalendar server controls. The following example shows the previous and next month text in red.</p>  <p>The example shows a calendar for May 2006. The navigation arrows (less than or equal to and greater than or equal to) are in red. The days of the week are listed in the header: Sun, Mon, Tue, Wed, Thu, Fri, Sat. The dates are displayed in a grid, with the first row starting on Monday (1st) and ending on Sunday (6th). The dates are: 1, 2, 3, 4, 5, 6; 7, 8, 9, 10, 11, 12, 13; 14, 15, 16, 17, 18, 19, 20; 21, 22, 23, 24, 25, 26, 27; 28, 29, 30, 31, and empty cells for the rest of the month.</p>
td.currentMonth	<p>The current month area of the calendar. Affects the Blog and BlogCalendar server controls. The following example shows the current month area in red.</p>  <p>The example shows a calendar for May 2006. The current month area (May 2006) is highlighted in red. The days of the week are listed in the header: Sun, Mon, Tue, Wed, Thu, Fri, Sat. The dates are displayed in a grid, with the first row starting on Monday (1st) and ending on Sunday (6th). The dates are: 1, 2, 3, 4, 5, 6; 7, 8, 9, 10, 11, 12, 13; 14, 15, 16, 17, 18, 19, 20; 21, 22, 23, 24, 25, 26, 27; 28, 29, 30, 31, and empty cells for the rest of the month.</p>
tr.dayNamesRow	<p>The day names row for the calendar. Affects the Blog and BlogCalendar server controls. The following example shows the day names area in red.</p>  <p>The example shows a calendar for May 2006. The day names row (Sun, Mon, Tue, Wed, Thu, Fri, Sat) is highlighted in red. The navigation arrows (less than or equal to and greater than or equal to) are in blue. The days of the week are listed in the header: Sun, Mon, Tue, Wed, Thu, Fri, Sat. The dates are displayed in a grid, with the first row starting on Monday (1st) and ending on Sunday (6th). The dates are: 1, 2, 3, 4, 5, 6; 7, 8, 9, 10, 11, 12, 13; 14, 15, 16, 17, 18, 19, 20; 21, 22, 23, 24, 25, 26, 27; 28, 29, 30, 31, and empty cells for the rest of the month.</p>

Style Class	Description
tr.weekRow td	<p>The rows in the calendar that contain each week's dates. Affects the Blog and BlogCalendar server controls. The following example shows the week row area in red.</p> 
td.inactiveDay	<p>The days with no associated blog posts. Affects the Blog and BlogCalendar server controls. The following example shows the days with no associated blog posts in red.</p> 
td.activeDay	<p>The days with an associated blog post. Affects the Blog and BlogCalendar server controls. The following example shows the days with an associated blog post in red.</p> 
Blogroll Styles	

Style Class	Description
div.blogSidebar h4	<p>The title area for the Blogroll. Affects the Blog and BlogRoll server controls. The following example shows the title area for the Blogroll in red.</p>  <p>Note: Changing this class, changes the title area for Categories and Recent Posts.</p>
div.blogRoll ul	<p>The link area for the Blogroll. Affects the Blog and BlogRoll server controls. The following example shows the link area for the Blogroll in red.</p> 
.blogrolllink	<p>The Blogroll link. Affects the Blog and Blogroll server controls. The following example shows the Blogroll link in red.</p> 
Blog Categories Styles	
div.blogSidebar h4	<p>The title area for the blog's Categories. Affects the Blog and BlogCategories server controls. The following example shows the title area for the blog categories in red.</p>  <p>Note: Changing this class changes the title area for Blogroll and Recent Posts.</p>

Style Class	Description
div.blogCategories ul	<p>The link area for the blog's Categories. Affects the Blog and BlogCategories server controls. The following example shows the link area for the blog's Categories in red.</p> 
.blogcatlink	<p>The blog's Categories links. Affects the Blog and BlogCategories server controls. The following example shows the blog's Categories links in red.</p> 
Blog Recent Posts Styles	
div.blogSidebar h4	<p>The title area for the blog's Categories. Affects the Blog and BlogRecentPosts server controls. The following example shows the title area for the blog categories in red.</p>  <p>Note: Changing this class changes the title area for Blogroll and Categories.</p>

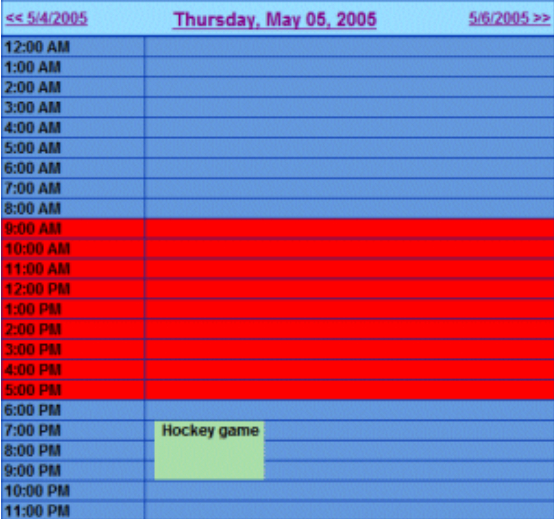
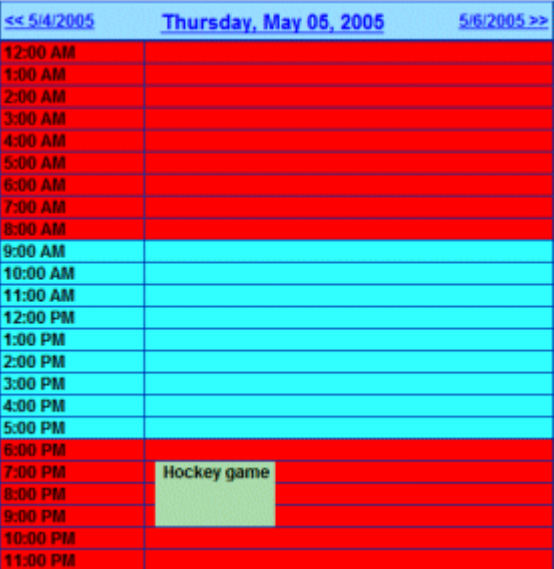
Style Class	Description
div.blogRecentPosts ul	<p>The link area for recent posts. Affects the Blog and BlogRecentPosts server controls. The following example shows the link area for recent posts in red.</p> 
.blogrecentlink	<p>The recent posts links. Affects the Blog and BlogCategories server controls. The following example shows the recent posts links in red.</p> 

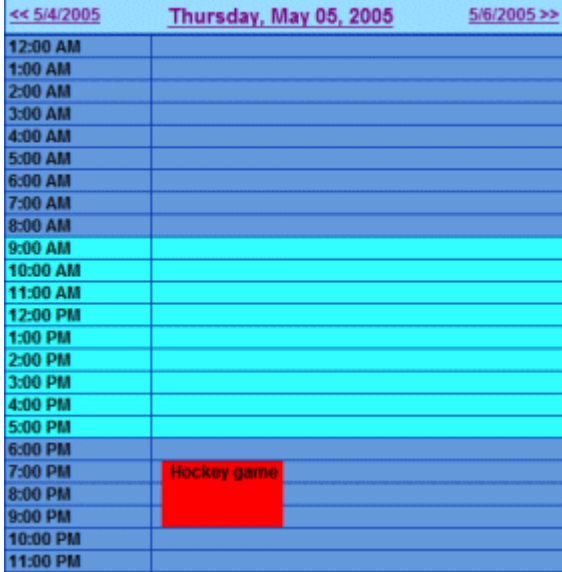
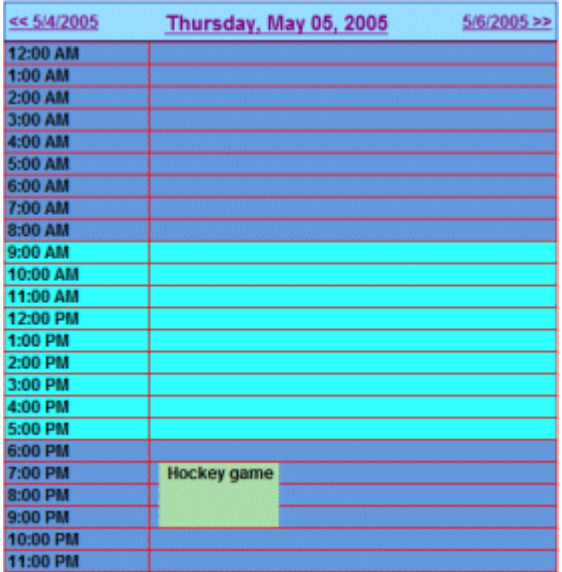
calendarStyles.css

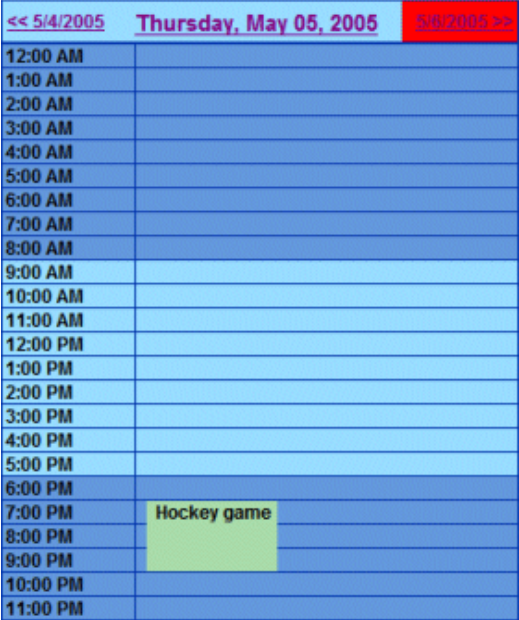

This file is located in `webroot/siteroot/Workarea/csslib`. It is used to customize the look of the Calendar server control.

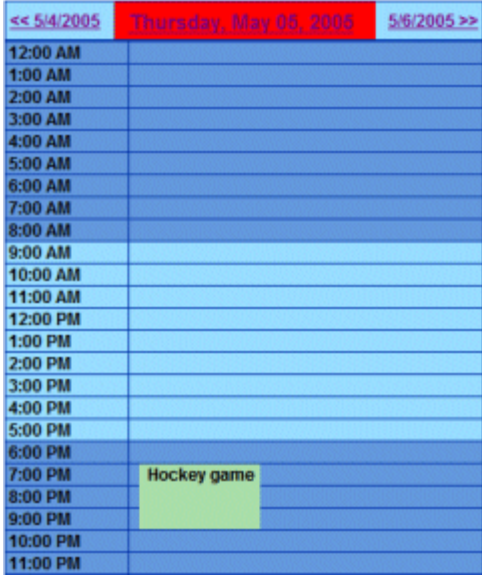
The table below contains the following sections.

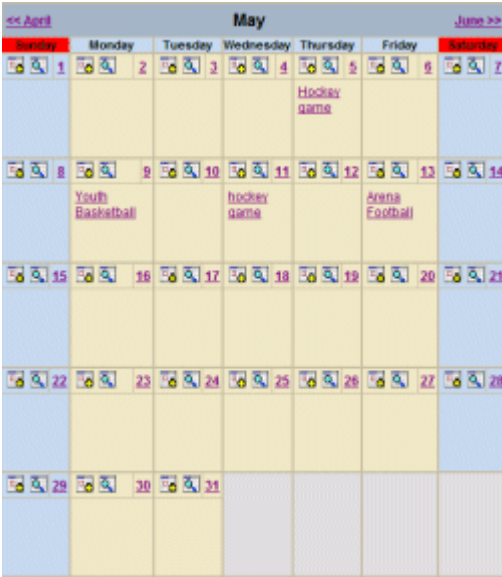
- "Single Day View (Day)" on page 913
- "Month View (month)" on page 916
- "Inline View (inline)" on page 928
- "Month-Up with Event (monthupwithevent)" on page 930
- "Event Type Control (used in month display)" on page 937


Style Class	Description	Attributes and Default Settings
Single Day View (Day)		
.dv_HourCell	<p>The workday hour cells. The following example shows these cells with a red background.</p> 	<ul style="list-style-type: none"> • background: #99DDFF • font-family: Arial, Arial Narrow, Times New Roman • font-size: 9pt • font-weight: bold
.dv_HourCellEve	<p>The evening and morning hour cells. The following example shows these cells with a red background.</p> 	<ul style="list-style-type: none"> • background: #6699DD • font-family: Arial, Arial Narrow, Times New Roman • font-size: 9pt • font-weight: bold



Style Class	Description	Attributes and Default Settings
.dv_EventTitle	<p>An event listed on the daily calendar. The following example shows the event with a red background.</p> 	<ul style="list-style-type: none"> • background: #aaddaa • font-family: Arial, Arial Narrow, Times New Roman • font-size: 9pt • font-weight: bold • cursor: pointer
.dv_BorderCell	<p>The borders of the hour cells. The following example shows the borders in red.</p> 	<ul style="list-style-type: none"> • background: #0033AA

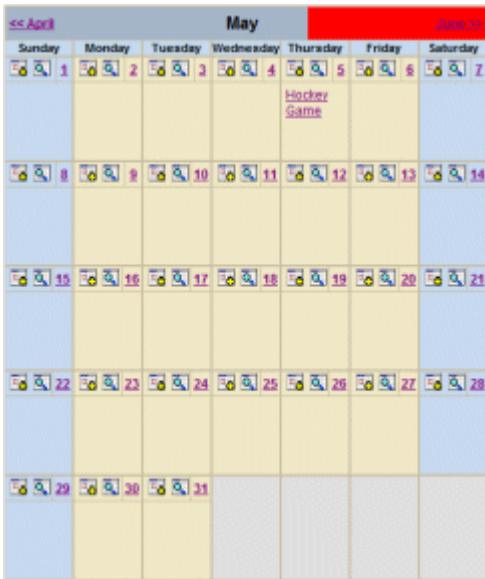
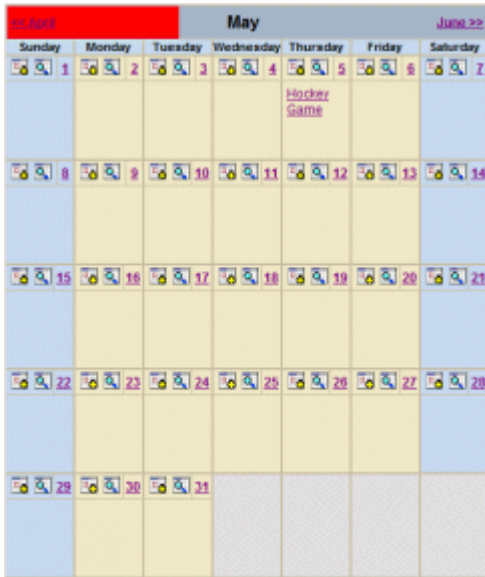
Style Class	Description	Attributes and Default Settings
.dv_DayHeaderNext	<p>The button for moving the calendar to the next day. The following example shows the day header next in red.</p> 	<ul style="list-style-type: none"> • background: #99DDFF • color: #0033AA • font-family: Arial • font-size: 9pt • font-weight: bold • text-align: right
.dv_DayHeaderPrev	<p>The button for moving the calendar to the previous day. The following example shows the day header previous in red.</p> 	<ul style="list-style-type: none"> • background: #99DDFF • color: #0033AA • font-family: Arial • font-size: 9pt • font-weight: bold • text-align: right



Style Class	Description	Attributes and Default Settings
.dv_DayHeader	<p>The day header for the calendar. The following example shows the day header in red.</p> 	<ul style="list-style-type: none"> • background: #99DDFF • color: #0033AA • font-family: Arial • font-size: 11pt • font-weight: bold • text-align: center
Month View (month)		

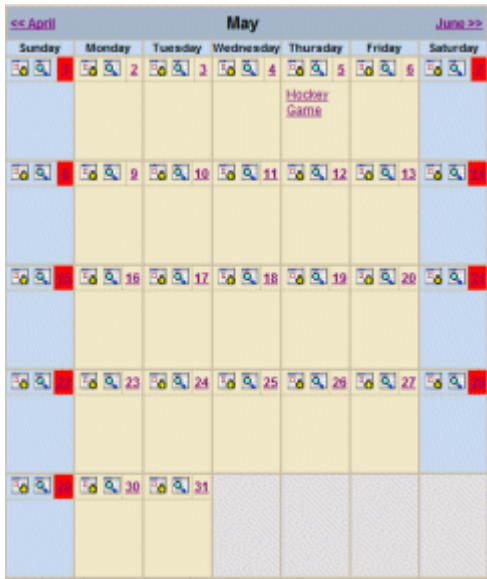

Style Class	Description	Attributes and Default Settings
.mv_weekendDOWHeader	<p>Weekend day of week header. The following example shows the weekend day of week header in red.</p>  <p>The screenshot shows a calendar for the month of May. The days of the week are listed at the top: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday. The dates 1 through 31 are displayed in a grid. The weekend days (Sunday and Saturday) are highlighted with a red background, and their corresponding day of the week headers are also in red. The text 'Hockey game' is visible on Thursday, May 5th, and 'Youth Basketball' is visible on Monday, May 8th. The text 'hockey game' is visible on Wednesday, May 11th, and 'Arena Football' is visible on Friday, May 13th.</p>	<ul style="list-style-type: none"> ● background: #c5daef ● font-family: arial ● font-weight: bold ● font-size: 8pt ● border-top: #c9bda5 1px solid ● border-bottom: #c9bda5 1px solid ● border-right: #c9bda5 1px solid ● border-left: #c9bda5 1px solid ● text-align: center



Style Class	Description	Attributes and Default Settings
.mv_weekdayDOWHeader	<p>Weekday day of week header. The following example shows the weekday day of week header in red.</p> 	<ul style="list-style-type: none"> • background: #c5daef • font-family: arial • font-weight: bold • font-size: 8pt • border-top: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid • border-right: #c9bda5 1px solid • border-left: #c9bda5 1px solid • text-align: center



Style Class	Description	Attributes and Default Settings
.mv_MonthHeader	<p>Month header. The following example shows the month header background in red.</p>  <p>The screenshot shows a calendar for the month of May. The header row, including the days of the week and the month name 'May', has a red background. The calendar grid shows dates from 1 to 31. Events are marked: 'Hockey game' on Thursday 5th, 'Youth Basketball' on Monday 8th, 'hockey game' on Wednesday 10th, and 'Arena Football' on Friday 12th. The background of the calendar grid is yellow, and the days of the week are labeled in blue.</p>	<ul style="list-style-type: none"> • font-weight: bold • font-size: 12pt • background: transparent • font-family: Arial
.mv_MonthHeaderBkg	<p>Month header background. The following example shows the month header background in red.</p>  <p>This screenshot is identical to the one above, showing a calendar for May with a red header background. It highlights the styling of the header background specifically.</p>	<ul style="list-style-type: none"> • border-right: #c9bda5 2px solid • border-top: #c9bda5 2px solid • border-left: #c9bda5 2px solid • font-weight: bold • font-size: 9pt • background: #A5B7C9 • font-family: arial



Style Class	Description	Attributes and Default Settings
.mv_MonthHeaderNext	<p>The button for moving the calendar to the next month. The following example shows the month header next background in red.</p> 	<ul style="list-style-type: none"> • font-weight: bold • font-size: 9pt • background: transparent • font-family: Arial • text-align: right
.mv_MonthHeaderPrev	<p>The button for moving the calendar to the previous month. The following example shows the month header previous background in red.</p> 	<ul style="list-style-type: none"> • font-weight: bold • font-size: 9pt • background: transparent • font-family: Arial • text-align: left



Style Class	Description	Attributes and Default Settings
.mv_NonMonthBorder	<p>The cells at the beginning or end of the month that do not contain days. The following example shows the non month border in red.</p>  <p>The image shows a calendar grid for May. The days of the week are listed at the top: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday. The dates 1 through 31 are shown in the grid. The cells for the first four days of May (1-4) are highlighted in red, indicating they are non-month border cells.</p>	<ul style="list-style-type: none"> • border-right: #c9bda5 1px solid • border-top: #c9bda5 1px solid • background: #e0e0e0 • border-left: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid
.mv_DateCell	<p>The cells where the numerical weekdays appear. The following example shows the date cell in red.</p>  <p>The image shows a calendar grid for May. The days of the week are listed at the top: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday. The dates 1 through 31 are shown in the grid. The cells for the first four days of May (1-4) are highlighted in red, indicating they are date cells.</p>	<ul style="list-style-type: none"> • font-weight: bold • font-size: 9pt • background: #F0E9C5 • border-left: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid • font-family: arial



Style Class	Description	Attributes and Default Settings
.mv_DateCellWeekend	<p>The cells where the numerical weekend days appear. The following example shows the date cell weekend in red.</p> 	<ul style="list-style-type: none"> • font-weight: bold • font-size: 9pt • background: #C5DAEF • border-left: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid • font-family: arial
.mv_DateCellToday	<p>The cells where today's numerical day appears. The following example shows the date cell today in red.</p> 	<ul style="list-style-type: none"> • background: #FFE354 • border-left: #b38b3b 1px solid • border-bottom: #b38b3b 1px solid • font-family: Arial • font-size: 9pt • font-weight: bold


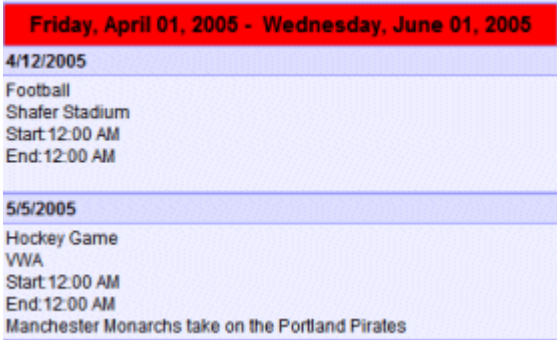
Style Class	Description	Attributes and Default Settings
<p><code>.mv_DateCellEmptyWeekend</code></p> <p>Note: These cells appear empty when you are not logged in to Ektron CMS400.NET.</p>	<p>The weekend cells that hold the Add Event and View Event buttons. The following example shows the date cell empty weekend in red.</p> 	<ul style="list-style-type: none"> • background: #C5DAEF • border-bottom: #c9bda5 1px solid • font-family: Arial • font-size: 9pt • font-weight: bold
<p><code>.mv_DateCellEmptyToday</code></p> <p>Note: This cell appears empty when you are not logged in to Ektron CMS400.NET.</p>	<p>The today cell that holds the Add Event and View Event buttons. The following example shows the date cell empty today in red.</p> 	<ul style="list-style-type: none"> • background: #FFE354 • border-bottom: #B38B3B 1px solid • font-family: Arial • font-size: 9pt • font-weight: bold

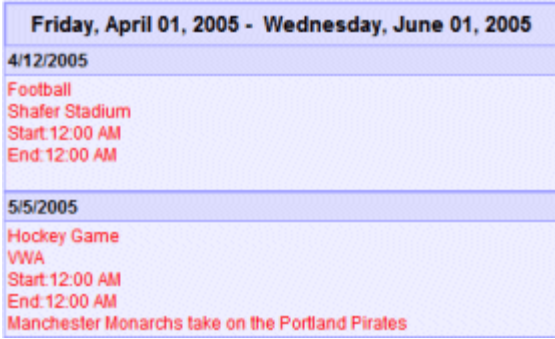
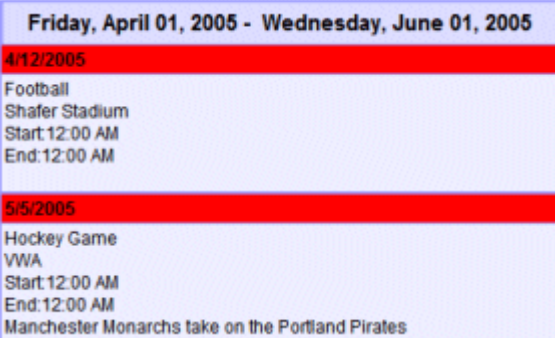
Style Class	Description	Attributes and Default Settings
<p><code>.mv_DateCellEmpty</code></p> <p>Note: These cells appear empty when you are not logged in to Ektron CMS400.NET</p>	<p>The weekday cells that holds the Add Event and View Event buttons. The following example shows the date cell empty in red.</p> 	<ul style="list-style-type: none"> • background: #f0e9c5 • border-bottom: #c9bda5 1px solid • font-family: Arial • font-size: 9pt • font-weight: bold
<p><code>.mv_eventContainer</code></p>	<p>The popup window that appears when the event is hovered over. The following example shows the event container circled in red.</p> 	<ul style="list-style-type: none"> • visibility: hidden • position: absolute <p>Note: When the visibility property is set to visible, the event container appears upon page load.</p>


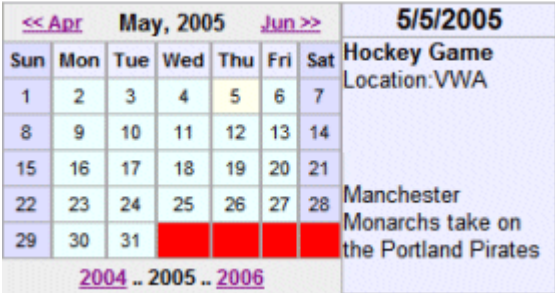
Style Class	Description	Attributes and Default Settings
.mv_dayBorderToday	<p>The today cell that holds listed events. The following example shows the day border today in red.</p> 	<ul style="list-style-type: none"> • border-right: #b38b3b 2px solid • border-top: #b38b3b 2px solid • background: #ffe354 • border-left: #b38b3b 2px solid • border-bottom: #b38b3b 2px solid
<p>.mv_dayBorder</p> <p><u>Note: This class affects all weekday cells, except the today cell.</u></p>	<p>The weekday cell that holds listed events. The following example shows the day border in red.</p> 	<ul style="list-style-type: none"> • border-right: #c9bda5 1px solid • border-top: #c9bda5 1px solid • background: #f0e9c5 • border-left: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid

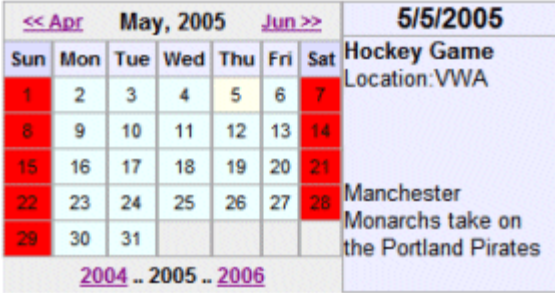
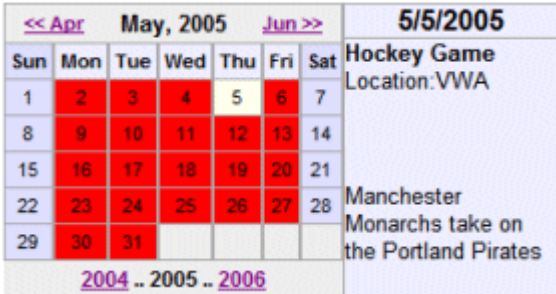
Style Class	Description	Attributes and Default Settings
.mv_dayBorderWeekend	<p>The Weekend day cell that holds listed events. The following example shows the day border weekend in red.</p> 	<ul style="list-style-type: none"> • border-right: #c9bda5 1px solid • border-top: #c9bda5 1px solid • background: c5daef • border-left: #c9bda5 1px solid • border-bottom: #c9bda5 1px solid
.mv_eventTitle	<p>The event title cell. The following example shows the event title cell in red.</p> 	<ul style="list-style-type: none"> • font-weight: normal • font-size: 9pt • background: transparent • font-family: Arial

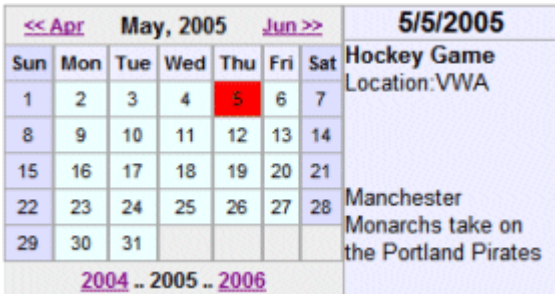
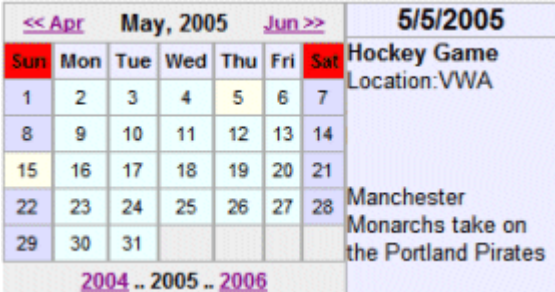
Style Class	Description	Attributes and Default Settings
.mv_DayCellToday	<p>Today's event title cell. The following example shows the day cell today in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-weight: bold • font-size: 9pt
.mv_TodayCell	<p>The font for the event pop-up. The following example shows the today cell font in red.</p> 	<ul style="list-style-type: none"> • font-weight: normal • font-size: 9pt • background: #C5DAEF • font-family: Arial

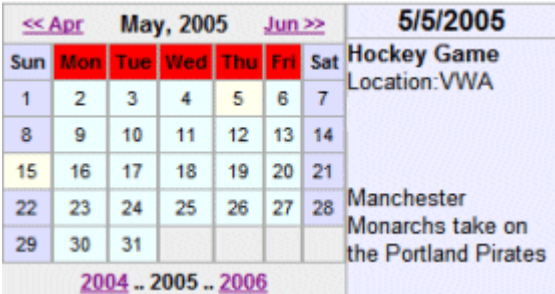
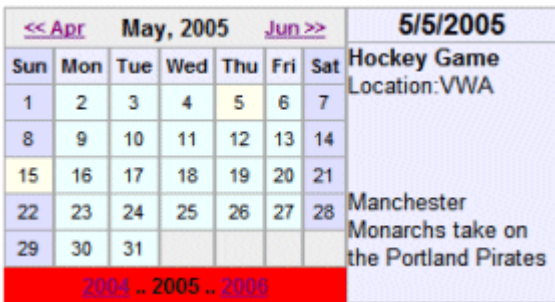
Style Class	Description	Attributes and Default Settings
.mv_eventPopup	<p>The border for the event pop-up. The following example shows the event pop-up border in red.</p> 	<ul style="list-style-type: none"> • border-right: #000000 1px solid • border-top: #000000 1px solid • border-left: #000000 1px solid • border-bottom: #000000 1px solid
Inline View (inline)		
.ILViewHeaderBkg	<p>The inline view header background. The following example shows the view header background in red.</p> 	<ul style="list-style-type: none"> • border-right: #9999ff 2px solid • border-top: #9999ff 2px solid • font-weight: bold • font-size: 11pt • background: #dddfdf • border-left: #9999ff 2px solid • font-family: Arial • text-align: center

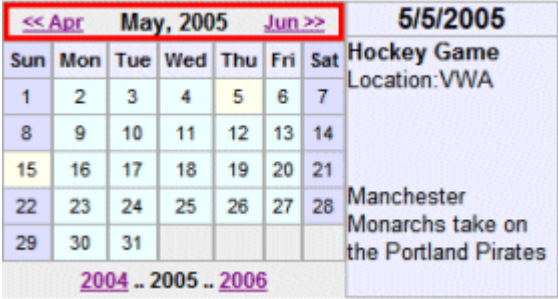
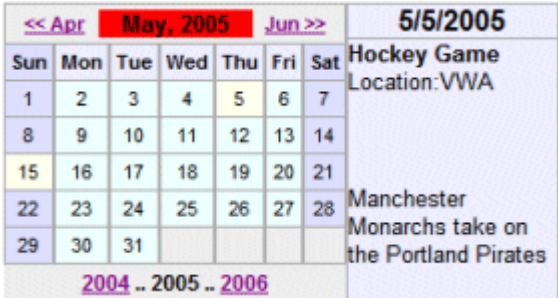
Style Class	Description	Attributes and Default Settings
.ILViewEvent	<p>Font for the event description. The following example shows the view event font in red.</p>  <p>The example shows two event entries. The first entry is for a football game at Shafer Stadium, starting and ending at 12:00 AM. The second entry is for a hockey game at VWA, also starting and ending at 12:00 AM. The text is red and includes the date range 'Friday, April 01, 2005 - Wednesday, June 01, 2005'.</p>	<ul style="list-style-type: none"> • font-size: 9pt • font-family: Arial
.ILViewDate	<p>The date for the inline view. The following example shows the view date background in red.</p>  <p>The example shows the same two event entries as the previous one. The date text is red and has a red background. The text is bold and includes the date range 'Friday, April 01, 2005 - Wednesday, June 01, 2005'.</p>	<ul style="list-style-type: none"> • border-right: #9999ff 1px solid • border-top: #9999ff 1px solid • font-weight: bold • font-size: 9pt • background: #dddddff • border-left: #9999ff 1px solid • font-family: Arial

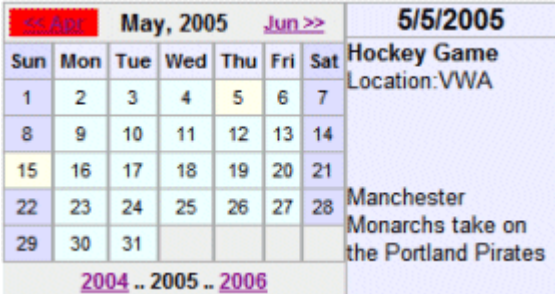
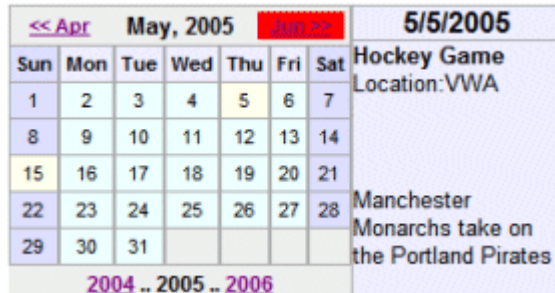

Style Class	Description	Attributes and Default Settings
.ILViewDayCell	<p>The cell the events are listed in. The following example shows the view day cell in red.</p> 	<ul style="list-style-type: none"> • border-right: #9999ff 1px solid • border-top: #9999ff 1px solid • font-weight: bold • font-size: 9pt • background: #eeeeff • border-left: #9999ff 1px solid • border-bottom: #9999ff 1px solid • font-family: Arial
Month-Up with Event (monthupwithevent)		
.mup_nonMonthCell	<p>The cells at the beginning or end of the month that do not contain days. The following example shows the non-month cells in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: normal • font-size: 8pt • text-align: center • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid

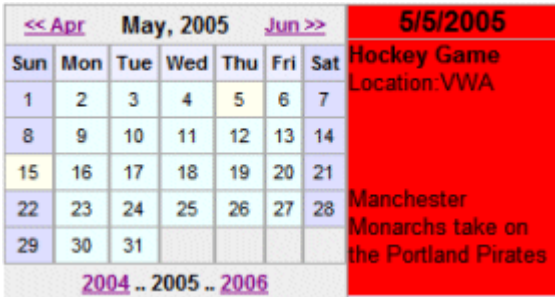
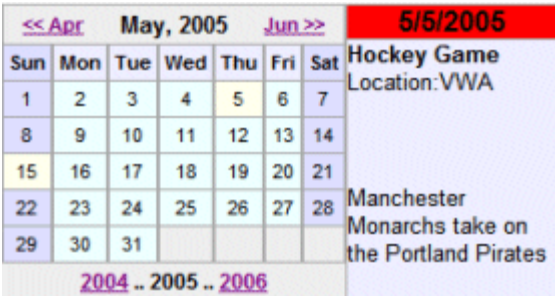
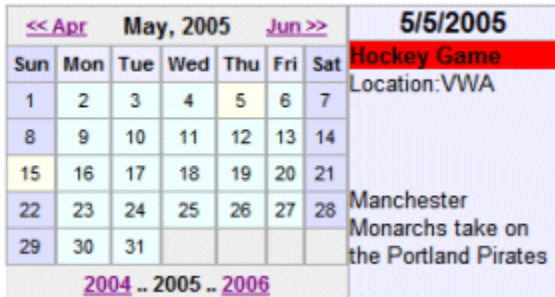
Style Class	Description	Attributes and Default Settings
.mup_weekendCell	<p>The weekend day cell. The following example shows the weekend cell in red.</p> 	<ul style="list-style-type: none"> • background: #dedeff • font-family: Arial • font-weight: normal • font-size: 8pt • text-align: center • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid
.mup_weekdayCell	<p>The week day cell. The following example shows the week day cell in red.</p> 	<ul style="list-style-type: none"> • background: #eeffff • font-family: Arial • font-weight: normal • font-size: 8pt • text-align: center • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid

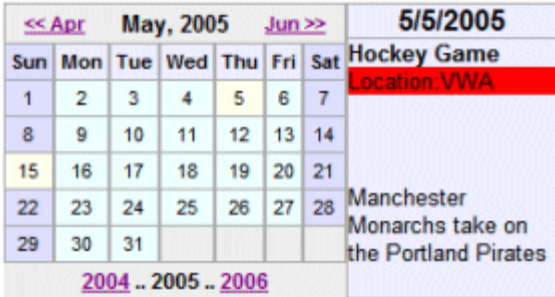
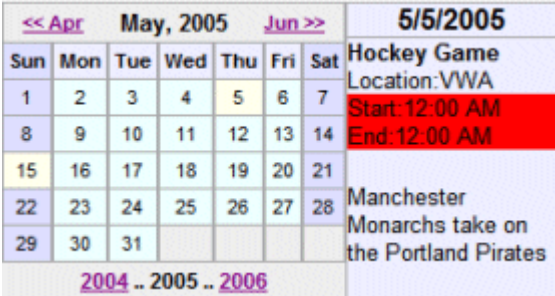
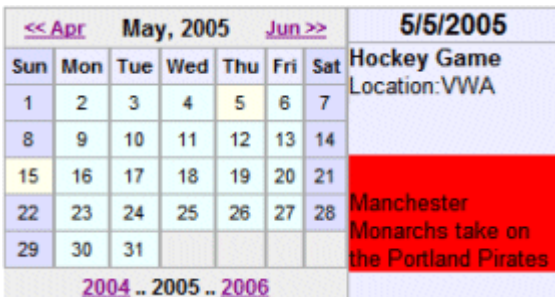
Style Class	Description	Attributes and Default Settings
.mup_weekdayEventCell	<p>Highlights the weekdays that have events. The following example shows the weekday event cell in red.</p> 	<ul style="list-style-type: none"> • background: #ffffee • font-family: Arial • font-weight: normal • font-size: 8pt • text-align: center • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid • cursor: pointer
.mup_weekendDOWHeader	<p>Weekend day of week header. The following example shows the weekend day of week header in red.</p> 	<ul style="list-style-type: none"> • background: #ddddff • font-family: Arial • font-weight: Bold • font-size: 8pt • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid • text-align: center



Style Class	Description	Attributes and Default Settings
.mup_weekdayDOWHeader	<p>Weekday day of week header. The following example shows the weekday day of week header in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeff • font-family: Arial • font-weight: Bold • font-size: 8pt • border-right: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid • text-align: center
.mup_yearFooter	<p>The year selection cell. The following example shows the year footer in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: Bold • font-size: 9pt • text-align: center



Style Class	Description	Attributes and Default Settings
.mup_monthHeaderBorder	<p>The border around the month header. The following example shows the month header border in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: Bold • font-size: 10pt • border-right: #aaaaaa 1px solid • border-top: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid • text-align: center
.mup_monthHeader	<p>The month header. The following example shows the month header in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: Bold • font-size: 10pt • text-align: center

Style Class	Description	Attributes and Default Settings
.mup_monthHeaderPrev	<p>The previous button, on the month header. The following example shows the month header previous back ground in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: Bold • font-size: 8pt • text-align: center
.mup_monthHeaderNext	<p>The next button, on the month header. The following example shows the month header next background in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeee • font-family: Arial • font-weight: Bold • font-size: 8pt • text-align: center
.mup_eventDispBkg <u>Note: .mup_eventDispBkg deals with the background layer of the event display. .mup_eventDisplay deals with the layer the event is displayed on.</u>	<p>The background for the event display. The following example shows the event display background in red.</p> 	<ul style="list-style-type: none"> • background: #eeeeff • border-right: #aaaaaa 1px solid • border-top: #aaaaaa 1px solid • border-bottom: #aaaaaa 1px solid • border-left: #aaaaaa 1px solid

Style Class	Description	Attributes and Default Settings
.mup_eventDisplay <u>Note: .mup_eventDispBkg deals with the background layer of the event display. .mup_eventDisplay deals with the layer the event is displayed on.</u>	<p>The event display. The following example shows the event display in red.</p> 	background: transparent
.mup_eventDisplayDate	<p>The event display date header. The following example shows the event display date in red.</p> 	<ul style="list-style-type: none"> • font-size: 12pt • font-weight: bold • background: transparent • font-family: Arial • text-align: center • border-bottom: #aaaaaa 1px solid
.mup_eventDisplayEventTitle	<p>Display for the event title. The following example shows the event display event title in red.</p> 	<ul style="list-style-type: none"> • font-weight: bold • background: transparent • font-family: Arial • font-size: 10pt • text-align: left

Style Class	Description	Attributes and Default Settings
.mup_eventDisplayLocation	<p>Display for the event location. The following example shows the event display location in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-weight: normal • font-size: 10pt • text-align: left
.mup_eventDisplayTimes	<p>Display for event times. The following example shows the event display times in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-weight: normal • font-size: 10pt • text-align: left
.mup_eventDisplayLongDesc	<p>Displays the event's long description. The following example shows the event display long description in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-weight: normal • font-size: 10pt • text-align: left
Event Type Control (used in month display)		

Style Class	Description	Attributes and Default Settings
.etCtrl_Background	<p>The background of the event type control cell. The following example shows the event type control background in red.</p> 	<ul style="list-style-type: none"> • border-right: #c9bda5 2px solid • border-top: #c9bda5 2px solid • border-left: #c9bda5 2px solid • border-bottom: #c9bda5 2px solid • font-size: 9pt • background: #A5B7C9 • font-family: Arial
.etCtrl_InstructCell	<p>The instructions cell for event type control. The following example shows the instruction cell in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-size: 9pt • font-weight: bold • text-align: right • vertical-align: top

Style Class	Description	Attributes and Default Settings
.etCtrl_SelCell	<p>The cell the selection box is displayed in. The instructions cell for event type control. The following example shows the selection cell in red.</p> 	<ul style="list-style-type: none"> • background: transparent • font-family: Arial • font-size: 9pt • text-align: left • vertical-align: top
.etCtrl_selectBox	<p>The event type selection box. The instructions cell for event type control. The following example shows the selection box in red.</p> 	<ul style="list-style-type: none"> • background: #c5daef

designformentry.css

Used internally by the Ektron CMS400.NET. It should not be changed.

pop_style.css

This file is located in `webroot/siteroot/Workarea/csslib`. It is used to customize the look of the DhtmlMenu server control. The following is a list of style sheet properties that can be modified.

Style Class	Description	Attribute	Default Value
.lowText	Text of the floating menu as displayed on the menu.	<ul style="list-style-type: none"> color font text-align text-decoration 	<ul style="list-style-type: none"> #000000 (Black) 12px Arial, sans-serif left none
.highText	Text of the floating menu when highlighted.	<ul style="list-style-type: none"> color font text-align text-decoration 	<ul style="list-style-type: none"> #000000 (Black) 12px Arial, sans-serif left none
.roothighText <u>Note: This class only works when called by the CssClass property. Use the Font property in the Appearance category to define root menu text.</u>	Text of the root menu as displayed on a Web page.	<ul style="list-style-type: none"> color font text-align text-decoration 	<ul style="list-style-type: none"> #000000 (Black) 16px Arial, sans-serif center none
.itemBorder	Border color and size of the menu.	<ul style="list-style-type: none"> border 	<ul style="list-style-type: none"> 1px solid #666666
.itemBorderBlank	Separator border color and size of the menu.	<ul style="list-style-type: none"> border 	<ul style="list-style-type: none"> 1px solid #CCCCDD

CMS400Demo.css

This file is located in `webroot/siteroot`. It is used to customize the look of the Ektron Medical Demo site.

Developer Styles.css

The `Styles.css` file is located in `webroot/siteroot/Developer`. It is a general style sheet provided for creating a new Web application project.

Default.css

This file is located in `webroot/siteroot/Developer`. It is the default CSS file for the Developer Web site ASPX files.

List Summary OrderedList.css

The `OrderedList.css` file is located in `webroot/siteroot/Developer/ListSummary`. It is used to format the `ListSummaryOrderedList.aspx` page.

List Summary unorderedlist.css

The `unorderedlist.css` file is located in `webroot/siteroot/Developer/ListSummary`. It is used to format an unordered List Summary.

Collection OrderedList.css

The `OrderedList.css` file is located in `webroot/siteroot/Developer/Collection`. It is used to format the `CollectionOrderedList.aspx` page.

Collection unorderedlist.css

The `unorderedlist.css` file is located in `webroot/siteroot/Developer/Collection`. It is used to format an unordered Collection.

CSSMenu.css

This file is located in `webroot/siteroot/Developer/Menu`. It is used to format the `cssmenu.aspx` page.

Search unorderedList.css

The `unorderedlist.css` file is located in `webroot/siteroot/Developer/Search`. It is used to format the `SearchOrderedList.aspx` page.

Discussion Board CSS Files

The following files are used to customize the look of the Discussion Board. These files are located in `webroot/siteroot/Workarea/threadeddisc/themes`.

- `Chrome.css`

- cool.css
- jungle.css
- modern.css
- royal.css
- slate.css
- standard.css
- techno.css
- theme.css

Additional Information

Supporting Documents

You can find more detailed information about Ektron CMS400.NET in the following manuals:

- Ektron CMS400.NET Developer Manual
- Ektron CMS400.NET Administrator Manual
- Ektron CMS400.NET User Manual
- Ektron CMS400.NET Quick Start Manual
- Ektron CMS400.NET Setup Manual

Support

If you are experiencing trouble with any aspect of Ektron CMS400.NET, please contact the Ektron Support Team via our Web Site at

http://www.ektron.com/support/cms400_support.aspx

Sales

For questions and answers about purchasing Ektron CMS400.NET from Ektron, contact the sales department by sending an e-mail to:

sales@ektron.com

Information

For general information about Ektron and its products, you may send an e-mail to:

info@ektron.com