

# Business

Ektron

### Ektron CMS400.NET Developer Reference Manual

Version 5.0, Revision 11

Ektron® Inc. 5 Northern Blvd., Suite 6 Amherst, NH 03031 Tel: +1 603-594-0249 Fax: +1 603-594-0258 Email: sales@ektron.com http://www.ektron.com

Copyright 2005 Ektron®, Inc. All rights reserved.

EKTRON is a registered trademark of Ektron, Inc.

Version 5.0, Revision 11, September 9, 2005

For updated versions of this manual, please visit: http://www.ektron.com/cms400-web-cms.aspx?id=1701#documentation

#### **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 nonexclusive 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 must be additional seats (see product pages for current price). The Basic Package are as follows:

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

Ektron CMS100 -- Licensed for five seats (5 named users) per URL.

Ektron CMS200 -- Licensed for ten seats (10 named users) per URL.

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

Ektron CMS400 -- 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.

Ektron eWebWP — Licensed for unlimited seats 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 decompiler 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 on 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, MERCHANTIBILITY, 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 - 2005 Ektron, Inc. All rights reserved.

LA10031, Revision 1.4

### What's New in the 5.0 Release

- The Ektron Dreamweaver functions now look and act like standard .NET server controls. You can also use Ektron Dreamweaver extensions with ASP, ColdFusion, JSP and PHP pages. See Also: "Dreamweaver Extension Support" on page 207
- New Documentation: "Creating a Web Site" on page 2. This section describes creating a Web site in Microsoft Visual Studio .NET 2003 and Macromedia Dreamweaver MX 2004.
- The ability to create web sites manually using Ektron's custom ASP functions. See Also: "Custom ASP Functions" on page 314
- The ability to create web sites manually using Ektron's custom ColdFusion functions. See Also: "Custom ColdFusion Functions" on page 423
- The ability to create web sites manually using Ektron's custom PHP functions. See Also: "Custom PHP Functions" on page 451
- The RssAggregator Server Control lets you consume and display an RSS feed on your Web site. See Also: "RssAggregator Server Control" on page 193
- The BreadCrumb Server Control creates a trail of Web pages for your site visitors to follow. See Also: "BreadCrumb Server Control" on page 197
- The ability to create web sites manually using Ektron's custom JSP functions. See Also: "Custom JSP Functions" on page 480

### **Summary Table of Contents**

What's New in the 5.0 Release	i
About This Manual	1
Creating a Web Site	2
Creating a Web Application Using Microsoft Visual Studio	.NET
Creating a Site Using Macromedia Dreamweaver MX 2004	47
Introduction to Ektron CMS400.NET Server Controls	24
Opening the Sample Project	25
Setting Up a Template	26
Making the Server Controls Available	27
Inserting Server Controls Using Drag and Drop	29
Inserting Server Controls Programmatically	33
Using Drag & Drop and Programmatically Together	51
Data Binding with Server Controls	53
Caching with Server Controls	59
List of Server Controls	61
Login Server Control	63
Placing a Login Button	63
Login Server Control Properties	64
LanguageSelect Server Control	65
LanguageSelect Server Control vs. LanguageAPI Server	Con-
trol	65
LanguageSelect Server Control Properties	66
LanguageAPI Server Control	67
LanguageSelect Server Control vs. LanguageAPI Server	Con-
trol	67
LanguageAPI Server Control Properties	67
Code Behind Only Properties	68
ContentBlock Server Control	70
Static Content Block	70
Dynamic Content Block	72
XML Content Block	75
Using the ContentBlock Server Control Programmatically	/ Ex-
ample	78

FormBlock Server Control	80
FormBlock Server Control Properties	81
Automatic versus Manual Generation of Form Tags	83
Search Server Control	84
Search Server Control vs. IndexSearch Server Control	85
Search Server Control Properties	85
Specify Search Control To Display Search Results on a	Sepa-
rate Page	93
IndexSearch Server Control	96
IndexSearch Server Control vs. Search Server Control	97
IndexSearch Server Control Properties	97
Using the IndexSearch Server Control Programmatically	'Ex-
ample	105
Programmatically Predefined General Search Results Re	<del>)</del> -
placed By Specific Search Results.	108
Turn Off Caching In Netscape Navigator and Mozilla Fire	fox
112	
ListSummary Server Control	113
ListSummary Server Control Properties	114
List Summary Example	118
Collection Server Control	119
Collection Server Control Properties	120
Using the Collection Server Control Programmatically E	xam-
pie Colondor Sonier Control	123
Calendar Server Control Properties	120
Calendar Server Control Properties	128
Monu Server Control	131
Menu Server Control vo. DhtmlMonu Server Control	100
Menu Server Control Vs. Dittilimenu Server Control	100
Ilsing DisplayXslt Samples	101
DhtmlMenu Server Control	177
Menu Server Control vs. DhtmlMenu Server Control	178
DhtmlMenu Server Control Properties	178
MetaData Server Control	185
MetaData Server Control vs. MetaDataList Server Contro	1185
MetaData Server Control Properties	
Using the MetaData Server Control	186
MetaDataList Server Control	188
MetaData Server Control vs. MetaDataList Server Contro	J189
MetaDataList Server Control Properties	189
RssAggregator Server Control	193
RSSAggregator Server Control Properties	194

Using the RssAggro	egator Server Control	5
BreadCrumb Server	Control	7
BreadCrumb Prope 199	rties Apply to Web Forms, Not Web Pages	<b>i</b> -
Types of BreadCrur	nb Server Controls19	9
BreadCrumb Serve	r Control Properties20 <sup>7</sup>	1
Using the BreadCru	Imb Server Control	3
Making a Content B 204	lock's Title Appear in the Breadcrumb Trai	il
Dreamweaver Extension	n Support207	7
Requirements		7
Preparing Dreamwe	aver for Ektron CMS400.NET Functions	•••
Installing the Exten	sions	8
Setup		3
Using the Ektron Cl	MS400.NET Extensions21	7
Inserting Custom F	unctions	6
XML Functions		4
Standard ASP.NET Meth	10ds307	7
Replacement of Sta	ndard Methods	7
Referencing Metho	ds in Visual Basic309	9
Adding a New Web	Form	9
Requirements for S	tandard Methods310	0
Standard Methods		1
Add Content		2
<b>Custom ASP Functions</b>		4
Login Function		7
Login Tag Example		8
Placing Your Login	Button	8
Content Block Funct	on319	9
Content Block Exar	nples	2
Add Content		9
Add Content Examp	ole	1
Search Functions		3
Search Tag Example	e	6
Search Tag Attribut	es	(
Search Display Funct	44 minute 244	4
Search Display Exa	mpie	0 6
Search Display Attr	indies	0

List Summary Function	350
List Summary Example	352
List Summary Attributes	353
XML List Summary Function	358
Archive List Summary Function	363
Single Summary Function	364
Single Summary Example	365
Single Summary Attributes	365
RSS Summary Function	368
RSS Summary Example	369
RSS Summary Attributes	372
Collection Function	374
Collection Examples	374
Collection Attributes	379
Creating Custom Display Functions	380
RSS Collection Function	387
XML Collection Function	389
Random Content	394
Random Summary	395
Event Calendar Function	396
Event Calendar Style Sheet	396
Event Calendar Sample	397
Calendar Function	399
Calendar Style Sheet	401
Event Calendar Sample	401
Form Function	403
MultiPurpose Function	406
Metadata Function	407
Creating an Item List	407
ecmMetadata Example	408
More ecmMetadata Examples	410
ecmMetadata Attributes	412
Show Site Language Function	415
DHTML Menus	416
Additional Documentation in Ektron Knowledge Base	416
Displaying a Menu on a Web Page	416
Customizing the Menu's Appearance	417
ek_Menu.js	418
XML Menu	421
Custom ColdFusion Functions	423
Login Function	426

Content Block Function	
Add Content	
Random Content Function	
Random Summary Function	
Search Function	
Search Display Function	
List Summary Function	
XML List Summary Function	
RSS List Summary Function	
Archive List Summary Function	
Single Summary Function	
Collection Function	
XML Collection Function	
RSS Collection Function	
Event Calendar Function	
Form Function	444
MultiPurpose Function	
Metadata Function	
DHTML Menus	448
XML Menu Function	
Language Select Function	450

Custom PHP Functions	451
Login Function	
Content Block Function	
Add Content	
Search Function	
Search Display Function	
List Summary Function	
Archive List Summary Function	
Single Summary Function	
XML List Summary Function	
RSS List Summary Function	
Collection Function	
XML Collection Function	
RSS Collection Function	
Random Content Function	
Random Summary Function	
Event Calendar Function	
Form Function	
MultiPurpose Function	
Metadata Function	
DHTML Menus	
XML Menu Function	

	Language Select Function	
	Custom API	479
Custo	om JSP Functions	480
	Calendar Function	
	Event Calendar Function	
	Collection Function	
	XML Collection Function	
	Random Content Function	
	Random Summary Function	
	RSS Collection Function	
	Add Content	
	Content Block Function	
	Form Function	
	Language Select Function	491
	Login Function	491
	DHTML Menus	
	XML Menu Function	
	Metadata Function	
	Search Function	
	Search Display Function	
	List Summary Function	
	XML List Summary Function	
	RSS List Summary Function	
	Single Summary Function	
Wob	Services	502
	Introduction to web Services	
	ws Components and CMS400.NET Architecture	
	Installation of Filos	
	Sotup Instructions	
	Providing Web Service Information in Your CMS	
	Creating a Page that Provides a Web Service	
	oreating a rage that rovides a web oervice	
Synd	icating Web Content as RSS	526
1 - C	What is RSS Syndication?	
	RSS Syndication in Ektron CMS400.NET	
	Syndicating via .NET	528
	Architecture of RSS Syndication	
	Creating Your Own aspx Collection Page	
	Creating Your Own aspx List Summary Page	
		-

Learning About Visual Studio .NET	531
Grid Layout vs. Flow Layout	531
Additional Information	533
Supporting Documents	
Support	
Sales	
Information	533

### **Detailed Table of Contents**

What's New in the 5.0 Release	i
About This Manual	1
Creating a Web Site	2
Creating a Web Application Using Microsoft Visual Stud	dio .NET
Creating a Web Project	1
Steps to Create a Web Project	4
Using Web Forms	6
Creating a Site Using Macromedia Dreamweaver MX 2	0047
Defining a Local Site	9
Steps to Define a Folder for Use as a Local Site	9
Building the Site Map in the Files Window	
Steps to Add a Home Page	
Steps to Create a Subfolder and Files	
Creating the Template	
Steps to Create a Template	
Applying the Template	
Steps to Apply the Template	
Testing and Viewing the Site	
Steps to lest and view the Site	23
Introduction to Ektron CMS400.NET Server Control	ls24
Opening the Sample Project	25
Setting Up a Template	26
Making the Server Controls Available	27
Removing the Server Controls	29
Updating the Server Controls	
Inserting Server Controls Using Drag and Drop	
Adding a Server Control to a Template	
Using CMS Explorer to Browse Your Ektron CMS400.NI 32	ET Site
Build the Project	33
Inserting Server Controls Programmatically	
Steps to Insert a Server Control Programmatically	

Step 2: Create an Instance of the New Control	35
Step 3: Set the Properties	36
Step 4: Call the Fill Method	37
Step 5: Use .text Property to Display the Control	37
Server Control Properties in Code Behind	38
Displaying Properties for a Control	38
Mutual Server Control Properties in Code Behind	39
Accessing Additional Properties	41
Accessing Items in an Array	47
Referencing the Page Property	47
Customizing Server Controls	48
Recognizing the Server Control Within the HTML	49
Recognizing the Server Control Within the Code Behind	49
Customizing the Server Control in the Code Behind	50
Troubleshooting Error Creating Control Message	50
Using Drag & Drop and Programmatically Together	51
Data Binding with Server Controls	53
Steps to Data Binding Using Drag and Drop Server Controls	54
Customizing the DataGrid Table View	57
Caching with Server Controls	59
List of Server Controls	61
Login Server Control	63
Placing a Login Button	63
Login Server Control Properties	64
	U++
LanguageSelect Server Control	
LanguageSelect Server Control vs. LanguageAPI Server	65 Con-
LanguageSelect Server Control vs. LanguageAPI Server	65 Con-
LanguageSelect Server Control vs. LanguageAPI Server trol	65 Con- 65
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties	65 Con- 65 66
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageSelect Server Control Properties LanguageAPI Server Control	65 Con- 65 66 67
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server	65 Con- 65 65 67 Con-
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageSelect Server Control Properties LanguageAPI Server Control vs. LanguageAPI Server trol	65 Con- 65 66 67 Con- 67
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control vs. LanguageAPI Server trol	65 Con- 65 66 67 Con- 67
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties	65 Con- 65 65 67 Con- 67 67 68
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control vs. LanguageAPI Server trol Code Behind Only Properties ContentBlock Server Control	65 Con- 65 66 67 Con- 67 67 67 68 70
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control Properties LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block	65 Con- 65 66 67 Con- 67 67 67 68 70 70
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties	65 Con- 65 66 67 Con- 67 67 67 67 67 67 67 67 
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties. LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties. ContentBlock Server Control Static Content Block. ContentBlock Server Control Properties. Dynamic Content Block	65 Con- 65 67 Con- 67 67 67 67 67 67 67 67 
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control Vs. LanguageAPI Server trol LanguageAPI Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties Dynamic Content Block	65 Con- 65 67 Con- 67 67 67 67 67 67 67 70 71 72 75
LanguageSelect Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties. LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties. Dynamic Content Block XML Content Block The OverrideXslt and DisplayXslt Properties	65 Con- 65 66 67 Con- 67 67 67 67 67 67 70 70 71 72 75 76
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control Properties LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties Dynamic Content Block XML Content Block The OverrideXslt and DisplayXslt Properties	65 Con- 65 66 67 Con- 67 67 67 67 67 67 70 71 71 72 75 76 v Ex-
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control LanguageSelect Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties Dynamic Content Block XML Content Block The OverrideXslt and DisplayXslt Properties Using the ContentBlock Server Control Programmatically ample	65 Con- 65 66 67 Con- 67 Con- 67 67 67 67 67 70 71 72 75 76 y Ex- 78
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties LanguageAPI Server Control Properties LanguageAPI Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties Dynamic Content Block The OverrideXslt and DisplayXslt Properties Using the ContentBlock Server Control Programmatically ample	65 Con- 65 67 Con- 67 Con- 67 67 67 67 67 70 70 71 72 75 y Ex- 78 80
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties. LanguageAPI Server Control LanguageAPI Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties ContentBlock Server Control Static Content Block ContentBlock Server Control Properties. Dynamic Content Block XML Content Block The OverrideXslt and DisplayXslt Properties Using the ContentBlock Server Control Programmatically ample	65 Con- 65 67 Con- 67 Con- 67 67 67 67 67 67 67 70 70 70 70 y Ex- 78 80
LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties. LanguageAPI Server Control Properties. LanguageAPI Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties. Code Behind Only Properties. Code Behind Only Properties. ContentBlock Server Control Static Content Block. ContentBlock Server Control Properties. Dynamic Content Block XML Content Block. The OverrideXslt and DisplayXslt Properties. Using the ContentBlock Server Control Programmatically ample FormBlock Server Control Properties.	65 Con- 65 67 Con- 67 Con- 67 67 67 67 67 67 67 67 70 71 72 75 76 y Ex- 78 80 81
Login Cerver Control Properties LanguageSelect Server Control vs. LanguageAPI Server trol LanguageSelect Server Control Properties. LanguageAPI Server Control vs. LanguageAPI Server trol LanguageAPI Server Control Properties Code Behind Only Properties. Code Behind Only Properties. ContentBlock Server Control Static Content Block. ContentBlock Server Control Properties. Dynamic Content Block XML Content Block. The OverrideXslt and DisplayXslt Properties Using the ContentBlock Server Control Programmaticall ample. FormBlock Server Control Properties Automatic versus Manual Generation of Form Tags.	65 Con- 65 66 67 Con- 67 67 67 67 67 67 70 70 71 72 75 76 y Ex- 78 80 81 83

Search Server Control vs. IndexSearch Server Control	85
Search Server Control Properties	85
Search Server Control Property Example	88
Text Box Examples	89
Button Image Source Examples	90
Search Button Text Examples	91
Display Examples	
ShowDate Examples	
Specify Search Control To Display Search Results on a	a Sepa-
rate Page	93
IndexSearch Server Control	
IndexSearch Server Control vs. Search Server Control	97
IndexSearch Server Control Properties	97
SearchParmXML Property	102
Using the IndexSearch Server Control Programmatica	ly Ex-
ample	105
Programmatically Predefined General Search Results	Re-
placed By Specific Search Results.	108
Turn Off Caching In Netscape Navigator and Mozilla Fi	refox
112	
ListSummary Server Control	113
ListSummary Server Control Properties	
OrderKey Property	117
List Summary Example	118
Collection Server Control	119
Collection Server Control Properties	120
Example of comNovigation Display	120
ecmNavigation XSL code	121 122
Example of ecmTeaser Display	122
ecmTeaser XSL code	122
Using the Collection Server Control Programmatically	Evam-
nlo	123
Calendar Server Control	126
Calendar Server Control Properties	120
Calendar Style Sheet	120 121
	<b>IJI</b>
Single Dev View (Dev)	102 122
Single Day view (Day)	132 126
	00 I ۱۸۵
Month-Un with Event (monthunwithevent)	140 150
Fyent Type Control (used in month display)	150 157
Manu Server Control	
	160
Manu Carvar Cantral va DhtmlManu Carvar Cantral	160
Menu Server Control vs. DhtmlMenu Server Control	160 160
Menu Server Control vs. DhtmlMenu Server Control Menu Server Control Properties	160 160 161

SampleMenu	163
SampleMenu XSL code	164
TreeMenu	166
DetmiMonu Server Control	168 477
Monu Server Control ve. DitmiMonu Server Control	170
DetmiMonu Server Control Properties	170
Using a Style Sheet	182
MetaData Server Control	185
MetaData Server Control vs. MetaDataList Server Control.	185
MetaData Server Control Properties	185
Using the MetaData Server Control	186
MetaDataList Server Control	188
MetaData Server Control vs. MetaDataList Server Control	189
MetaDataList Server Control Properties	189
RssAggregator Server Control	193
RSSAggregator Server Control Properties	194
Using the RssAggregator Server Control	195
BreadCrumb Server Control	197
BreadCrumb Properties Apply to Web Forms, Not Web Pag	jes
199	
Types of BreadCrumb Server Controls Properties that Determine a Breadcrumb Trail's Appearance Properties that Determine How Pages Associated with the Form Appear on a Breadcrumb Trail	<ul><li><b>199</b></li><li>199</li><li>200</li><li>201</li></ul>
Using the BreadCrumb Server Control	201
Making a Content Block's Title Appear in the Breadcrumb 1 204	zus Frail
Steps to Use a Content Block's Title as a Breadcrumb	205
Dreamweaver Extension Support2	207
Requirements	207
Preparing Dreamweaver for Ektron CMS400.NET Functions 208	\$
Installing the Extensions	208
Setup	213
Using the Ektron CMS400.NET Extensions	217
Custom Function Symbols	219
Ektron Civice Command Menus	221
CMS Function	226
Accessing CMS Content Functions	226
Login	227

Inserting a Content Block	230
Inserting a Dynamic Content Block	234
Inserting a Collection	236
Inserting a List Summary	240
Inserting a Metadata Function for ASP.NET pages	245
Inserting a Metadata Function for ASP, CF, JSP or PHP page	s 248
Inserting a Search Function	251
Inserting a Search Display Function	257
Inserting a Single Summary Function	259
Inserting a Random Content Function	262
Inserting a Random Summary Function	264
Inserting an Event Calendar Function	266
Inserting a Form	269
Inserting a Multipurpose content block	273
Inserting an Advanced Calendar Tag	274
Inserting an XML Index Search	276
Inserting Display By Meta Value	282
Inserting a Standard Menu	287
Inserting a DHTML Menu	290
Inserting a Language Select Box	296
Inserting a Language API	298
Inserting an RSS Aggregator	301
XML Functions	304
	304
Accessing XML Function Commands	
Insert Content XSLT Tag	304
Insert Content XSLT Tag	304
Standard ASP.NET Methods	304
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic	304 307 307 307
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form	304 307 307 309 309
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods	304 307 307 309 309 309 309
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods Standard Methods	304 307 307 309 309 310 310
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods Standard Methods	304 307 307 309 310 311
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods Standard Methods Add Content	304 307 307 309 310 311 312
Accessing XML Function Commands. Insert Content XSLT Tag	304 307 307 309 310 311 312 313
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods Standard Methods Add Content Add Content Example	304 307 307 309 310 311 313 314
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods Replacement of Standard Methods Referencing Methods in Visual Basic Adding a New Web Form Requirements for Standard Methods Standard Methods Add Content Add Content Example Custom ASP Functions Login Function	304 307 307 309 310 311 312 313 314
Accessing XML Function Commands Insert Content XSLT Tag Standard ASP.NET Methods	
Accessing XML Function Commands. Insert Content XSLT Tag	304 307 307 309 310 311 312 313 314 317 318 318
Accessing XML Function Commands	304 307 307 309 310 311 312 313 314 317 318 318 318
Accessing XML Function Commands	304 307 307 309 310 311 312 313 314 317 318 318 319
Accessing XML Function Commands	

Dynamic and Static Combination	
Add Content	
Add Content Example	
Search Functions	
Search Tag Example	
Search Tag Attributes	
Starting Folder	
Recursive	
Target Page	
Text Box	
Maximum Characters	
Button Image Source	
Button Text	
Font Face	
Font Size	
Horizontal	
Spare 1	
Search Display Function	
Search Display Example	
Search Display with Default Content	
Search Display Attributes	
Max Number of Content Blocks Returned	
Style Info	
Snow Date	
List Summary Function	
List Summary Example	
List Summary Attributes	
Starting Folder	
Recursive	
Show Summary	
Order By	
Ascending/Descending	
Show Info	
Max Number	
XML List Summary Function	
Archive List Summary Function	363
Single Summary Function	364
Single Summary Example	265
Single Summary Attributes	
Content ID	
Show Summary	200
Style Info	366
Show Info	
Spare 1	
RSS Summary Function	
· · · · · · · · · · · · · · · · · · ·	

RSS Summary Example	369
RSS Summary Attributes	372
Starting Folder	372
Recursive	372
Order By	372
Ascending/Descending	373
Max Number	373
Collection Function	374
Collection Examples	374
The ecmNavigation Display Function	375
The ecmTeaser Display Function	376
Collection Attributes	379
ID	379
DisplayFunction	379
Creating Custom Display Functions	380
Function Structure	380
Creating a Basic Display Function	380
Building Blocks	383
Defining Display Functions	385
Dising a Custom Display Function	300 <b>207</b>
KSS Collection Function	30/
XWL Collection Function	389
Random Content	394
Random Summary	395
Random Summary Event Calendar Function	395 396
Random Summary Event Calendar Function Event Calendar Style Sheet	395 396 396
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample	395 396 396 397
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function.	395 396 396 397 399
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet	395 396 396 397 399 401
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample	395 396 396 397 399 401 401
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function	395 396 396 397 397 399 401 403
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function	395 396 396 397 399 401 401 403 406
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function	395 396 396 397 399 401 401 403 406 407
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List	395 396 396 397 399 401 401 403 406 407 407
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function. Creating an Item List ecmMetadata Example	395 396 396 397 399 401 401 403 406 407 407
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples	395 396 396 397 399 401 401 403 406 407 407 408 410
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function. Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag	395 396 397 397 399 401 401 403 406 407 407 408 410 411
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function. Creating an Item List ecmMetadata Example Using a Dynamic Metadata Tag ecmMetadata Attributes	395 396 396 397 399 401 403 406 407 407 408 410 411 411
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function. Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag ecmMetadata Attributes	395 396 396 397 399 401 401 403 406 407 407 408 410 411 412 412
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag Item List Spare 1	395 396 396 397 399 401 401 401 403 407 407 407 418 412 412 414
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag ecmMetadata Attributes Item List Spare 1 Show Site Language Function	395 396 397 397 399 401 401 403 406 407 407 408 410 411 412 414 414
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag Item List Spare 1 Show Site Language Function	395 396 397 399 401 401 403 406 407 408 410 412 412 412 414 415 416
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag ecmMetadata Attributes Item List Spare 1 Show Site Language Function DHTML Menus	395 396 396 397 399 401 401 403 406 407 408 410 412 412 415 416 416
Random Summary Event Calendar Function Event Calendar Style Sheet Event Calendar Sample Calendar Function Calendar Style Sheet Event Calendar Sample Form Function MultiPurpose Function Metadata Function Creating an Item List ecmMetadata Example More ecmMetadata Examples Using a Dynamic Metadata Tag ecmMetadata Attributes Item List Spare 1 Show Site Language Function DHTML Menus Additional Documentation in Ektron Knowledge Base Displaying a Menu on a Web Page	395 396 397 397 399 401 401 401 403 407 407 407 407 407 416 416 416 416

GenerateMenu and DropMenu	416
Displaying Menus in a Multi-language System	417
Customizing the Menu's Appearance	
ek_Menu.js	418
FontFamily	419
menuLiteBgColor	419
XML Menu	421
Custom ColdFusion Functions	423
Login Function	426
Login Tag Example	426
Content Block Function	
Static Content Block	427
Dynamic Content Block	427
XML Content Block	427
Add Content	
Add Content Example	428
Random Content Function	
Random Summary Function	
Search Function	
Search Tag Example	430
Search Display Function	431
Search Display Example	431
List Summary Function	431
List Summary Example	432
XML List Summary Function	
XML List Summary Example	433
RSS List Summary Function	
List Summary Example	434
Archive List Summary Function	434
Single Summary Function	434
Single Summary Example	435
Collection Function	435
Example Using the ecmNavigation Display Function	436
Example Using the ecmTeaser Display Function	438
Collection Variables	
DisplayFunction	
Building Blocks	
XML Collection Function	
RSS Collection Function	
Event Calendar Function	
Event Calendar Sample	
Form Function	
MultiPurpose Function	
Metadata Function	

DHTML Menus	
Additional Documentation in Ektron Knowledge Base	448
Displaying a Menu on a Web Page	448
GenerateMenu and DropMenu	
Customizing the Menu's Appearance	
ek_Menu.js	449
XML Menu Function	
Language Select Function	450
Custom PHP Functions	451
Login Function	
Login Function Example	455
Content Block Function	
Static Content Block	
Dynamic Content Block	456
XML Content Block	456
Add Content	
Add Content Example	457
Search Function	
Search Function Example	458
Search Display Function	
Search Display Function Example	459
List Summary Function	
List Summary Example	460
Archive List Summary Function	
Single Summary Function	
Single Summary Example	461
XML List Summary Function	
RSS List Summary Function	
Collection Function	
Collection Examples	
Example Using the ecmNavigation Display Function	463
Example Using the ecmTeaser Display Function	464
Collection Variables	466
DisplayFunction	466
Building Blocks	466
XML Collection Function	
RSS Collection Function	
Random Content Function	
Random Summary Function	
Event Calendar Function	
Event Calendar Sample	
Form Function	
MultiPurpose Function	
Metadata Function	

DHTML Menus	
Additional Documentation in Ektron Knowledge Base	476
Displaying a Menu on a Web Page	476
GenerateMenu and DropMenu	476
Customizing the Menu's Appearance	477
ek_Menu.js	477
XML Menu Function	
Language Select Function	
Custom API	479
Custom JSP Functions	480
Calendar Function	
Calendar Sample	483
Event Calendar Function	
Event Calendar Sample	484
Collection Function	
Collection Sample	485
XML Collection Function	
Random Content Function	
Random Summary Function	
RSS Collection Function	
Add Content	
Add Content Example	488
Content Block Function	
Static Content Block	488
Dynamic Content Block	489
XML Content Block	489
Form Function	
Language Select Function	
Login Function	
Login Function Example	492
DHTML Menus	
Additional Documentation in Ektron Knowledge Base	493
Displaying a Menu on a Web Page	493
GenerateMenu and DropMenu	493
Customizing the Menu's Appearance	
ek_ivienu.js	
Metadata Function	
Search Function	
Search Function Example	
Search Display Function	
Search Display Function Example	
List Summary Function	
List Summary Example	

XML List Summary Function	50(
RSS List Summary Function	501
Single Summary Function	501
Single Summary Example	50 <sup>-</sup>
Web Services	502
Introduction to Web Services	
CMS400.NET's Implementation of Web Services	503
Benefits of Using Web Services	503
WS Components and CMS400.NET Architecture	503
Software and Hardware Requirements	
Installation of Files	504
Setup Instructions	50
Providing Web Service Information in Your CMS	500
CMS400 NET Methods	50
The CMS400 NET Content Sample Page	510
The Search Display Method Parameters	
Sample Output from One Method	51
GetContentBlock	51
The News.aspx Sample (non-XML)	516
Identifying the Code Behind File	516
Setting Up an Instance of Web Services	517
Getting Content Blocks	517
The News.aspx Sample (XML)	519
Setting Up Instances of Web Services	519
Getting Content Blocks	519
Creating a Page that Provides a Web Service	524
Syndicating Web Content as RSS	526
What is RSS Syndication?	526
The RSS Specification	526
RSS Syndication in Ektron CMS400.NET	
Syndicating via NFT	528
Architecture of RSS Syndication	529
	529
Retrieving a Collection or List Summary as an RSS Feed	52
Creating Your Own aspx Collection Page	528
Example	529
Remarks	529
Example	529
Creating Your Own aspx List Summary Page	
Example	53(
Remarks	53(
Example	530

Learning About Visual Studio .NET	531
Grid Layout vs. Flow Layout	531
Additional Information	533
Supporting Documents	
Support	
Sales	
Information	533

### **About This Manual**

This manual can be used as a reference for the developer, or the person is 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 24.

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

- "Standard ASP.NET Methods" on page 307
- "Custom ASP Functions" on page 314
- "Custom ColdFusion Functions" on page 423
- "Custom PHP Functions" on page 451

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 .NET or by using the Dreamweaver extension supplied with the installation. See "Dreamweaver Extension Support" on page 179 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 502
- "Syndicating Web Content as RSS" on page 526

### **Creating a Web Site**

When creating a Web site Ektron recommends using one of the following programs.

- Microsoft Visual Studio .NET
- Macromedia Dreamweaver MX 2004

This chapter provides an overview of creating a site using each of these programs. Each product has its own niche. Microsoft Visual Studio .NET with its intellisense, separate pages for coding, Server Controls, and ease of database manipulation is geared more toward the programmer/developer.

On the other hand, Macromedia Dreamweaver MX 2004 with its easy to use customizable interface, state of the art design tools, and enhanced coding tools, is a streamlined design tool for the designer/developer.

For more information on creating a Web site using Microsoft Visual Studio .NET, see "Creating a Web Application Using Microsoft Visual Studio .NET" on page 2

For more information on creating a Web site using Macromedia Dreamweaver MX 2004, see "Creating a Site Using Macromedia Dreamweaver MX 2004" on page 7.

## Creating a Web Application Using Microsoft Visual Studio .NET

**NOTE** Some information in this section has been copied from various Microsoft Web sites. These Web sites include:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vbcon/html/ vbwlkwalkthroughcreatingbasicwebformspage.asp

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dndotnet/html/ introwebforms.asp http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vbcon/html/vbtskCreatingWebProject.asp

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vbcon/html/vbconintroductiontowebprojects.asp

When using Microsoft Visual Studio .NET, you are creating a Web application instead of a Web site. Web Applications typically involve more programming code and can include Web services. Fundamentally Web applications and Web sites can be the same thing. Microsoft's view is Visual Studio .NET is for building applications, Web or otherwise. If you are building a simple web site, their recommendation is to use Microsoft FrontPage.

NOTE

Ektron recommends using Microsoft Visual Studio .NET or Macromedia Dreamweaver MX 2004. Ektron has added extended functionality for both products in Ektron CMS400.NET.

Most Web applications include the following:

- a Web project
- references
- assembly information
- a Global.asax file
- a style sheet
- a web.config file
- one or more Web Forms

For a complete description of the above items, visit the following Microsoft Web site:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/ vbcon/html/vbconintroductiontowebprojects.asp

In this section, the following is explained:

- "Creating a Web Project" on page 4
- "Using Web Forms" on page 6

This section does not explain the use of **Ektron CMS400.NET** Server Controls when building a Microsoft Visual Studio .NET Web Application. For information on **Ektron CMS400.NET** Server Controls, see "Introduction to Ektron CMS400.NET Server Controls" on page 24.

**IMPORTANT Ektron CMS400.NET** Server Controls are an integral part of using **Ektron CMS400.NET** and Microsoft Visual Studio .NET together. After completing this section read, "Introduction to Ektron CMS400.NET Server Controls" on page 24.

#### **Creating a Web Project**

The Web server on which your project exists can be on either a local or remote computer.

#### Steps to Create a Web Project

- Click File > New > Project to display the New Project dialog box.
- In the Project Types pane, select either Visual Basic or Visual C#.
- 3. In the Templates pane, select **ASP.NET Web Application**.
- In the Location text box, specify the URL where you want to place the Web project, including the name you want to give it.

For example, to create a project titled **MyFirstWebApp** at the default localhost location, Insert the following URL string in the Location text box:

http://localhost/MyFirstWebApp

This allows for exact placement of your project files in a specific directory on a Web server.

New Project		
Project Types:	Templates:	
Visual Basic Projects		
Visual J# Projects  Visual C++ Projects	Windows Class Library Windows Application Control Library	
Setup and Deployment Projects     Other Projects     Visual Studio Solutions		
	Smart Device ASP.NET Web Application ASP.NET Web Service	
A project for creating an application with a Web user	rinterface	
Name: MyFirstWebApp		
Location: http://localhost/MyFirstWeb	App	
Project will be created at http://localhost/MyFirstWebApp.		
<b>▼</b> Mor <u>e</u>	OK Cancel Help	

NOTE

**TE** If the local computer you are working on is also a Web server, you can create your Web project on that computer. The first time you start Visual Studio, the default location of a Web project is your local computer. In subsequent Visual Studio sessions, you can select the location of a previously used Web server from a drop-down list. If you want to choose a different server, you can click the Browse button to find other servers on your network, or you can type in the URL to a server (for example, http://domain.com) and press ENTER.

5. Click **OK** to create the Web project.

Visual Studio connects to the server and adds the appropriate project items to Solution Explorer in the project node.

$\triangleleft \triangleright \mathbf{x}$	Solution Explorer - MyFirstWebApp	×
:::: 🔼		
	Solution 'MyFirstWebApp' (1 project)	
	📄 👰 MyFirstWebApp	
	The forences	
	AssemblyInfo.vb	
	👩 Global.asax	
	A Styles.css	
	💀 Web.config	
	WebForm1 aspx	
	Webi offinitiaspx	
:::::		T
	Solution Expl 🖄 Class View   🤐 Search	
	Properties 4 3	×
	WebForm1 acry File Properties	7
	Webronninaspx File Properties	
	Advanced	

#### **Using Web Forms**

Web Forms and ASP.NET were created to overcome some of the limitations of ASP. These new strengths include:

- Separation of HTML interface from application logic
- A rich set of server-side controls that can detect the browser and send out appropriate markup language such as HTML
- Less code to write due to the data binding capabilities of the new server-side .NET controls
- Event-based programming model that is familiar to Microsoft Visual Basic® programmers

 Allows third parties to create controls that provide additional functionality. For example, the Ektron CMS400.NET Server Controls. See Also: "Introduction to Ektron CMS400.NET Server Controls" on page 24

On the surface, Web Forms seem just like a workspace where you draw controls. In reality, they can do a whole lot more. But normally you will just place any of the various controls onto the Web Form to create your UI. The controls you use determine which properties, events, and methods you will get for each control.

There is an excellent step by step explanation of creating and using Web Forms located on the following Web page:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/ dndotnet/html/introwebforms.asp

### Creating a Site Using Macromedia Dreamweaver MX 2004

NOTE

The information in this section has been copied from the Macromedia Web site. It appears on the following Web page:

http://www.macromedia.com/cfusion/knowledgebase/index.cfm?id=tn\_14567

The information has been modified by Ektron, Inc. to discuss .NET instead of Coldfusion and has been updated to reflect Macromedia Dreamweaver MX 2004 instead of Macromedia Dreamweaver 3.0

You can easily create a web site with Macromedia Dreamweaver MX 2004. This section discusses:

- creating a site
- creating files
- creating a template
- applying the template
- viewing the site

This chapter does not explain the use of **Ektron CMS400.NET** functions when building a Dreamweaver site. For Information on **Ektron CMS400.NET** functions in Dreamweaver, see "Dreamweaver Extension Support" on page 207.

**IMPORTANT Ektron CMS400.NET** Dreamweaver functions are an integral part of using **Ektron CMS400.NET** and Dreamweaver together. After completing this section, read "Dreamweaver Extension Support" on page 207.

The process of creating a basic site in Macromedia Dreamweaver MX 2004 can be broken down into the following six segments.

- "Defining a Local Site" on page 9
- "Building the Site Map in the Files Window" on page 12
- "Creating a Subfolder and Files" on page 17
- "Creating the Template" on page 17
- "Applying the Template" on page 23
- "Testing and Viewing the Site" on page 23

By completing the examples in these sections, you create a site with a template, hyperlinks and multiple pages. When you are completed, your site should look similar to the picture below. You might choose different colors and text but the layout should be the same.



#### **Defining a Local Site**

Before creating pages for a site, it is necessary to define the site in Dreamweaver. This is done by selecting a folder on the computer's hard disk in which all the files for the site will be contained. This is the local root folder.

After the site is defined, the pages for the site are created in Dreamweaver, saved inside the local root folder, and then uploaded to the Web server. The local site and the remote site on the server should have exactly the same structure.

**NOTE** If Dreamweaver is used to upload the site, it automatically replicates the file structure of the local root folder on the server.

#### Steps to Define a Folder for Use as a Local Site

To define the folder as the local root folder in Dreamweaver, do the following:

1. On the welcome screen click Dreamweaver Site...



Or

Click Site > Manage Site > New > Site

2. Enter the name of your site in the text box.

A site, in Macromedia Dreamweaver MX 2004, is a collection of files and folders that corresponds to a website on a server.

What would you like to name your site?

MyFirstSite

Example: mySite

- 3. Click Next.
- 4. Click the Yes, I want to use server technology radio button.
- 5. Choose **ASP.NET** from the **which server technology** dropdown box.

Do you want to work with a server technology such as ColdFusion, ASP.NET, ASP, JSP, or PHP?

No, I do not want to use a server technology.

Yes, I want to use a server technology.

Which server technology?

ASP.NET VB

6. Click Next.

- Choose how you want to work with your files during development. For these example steps choose the Edit and test locally (my testing server is on this computer) radio button.
- 8. Choose where you want to store the files.

How do you want to work with your files during development?
Edit and test locally (my testing server is on this computer)
Edit locally, then upload to remote testing server
Edit girectly on remote testing server using local network

Where on your computer do you want to store your files?
c:\inetpub\wwwroot\MyFirstSite
Because IIS has been installed on your computer, your computer can be used as a local testing

Because IIS has been installed on your computer, your computer can be used as a local testing server.
9. Click Next.

10. Add the URL to browse to the root of your site.

Dreamweaver communicates with your testing server using HTTP (just like a browser), so it needs to know the URL of your site's root folder.

What URL would you use to browse to the root of your site?

http://localhost/MyFirstSite/

Example: http://ServerOne/RootFolder/

Test URL

11. Click **Test URL**. If everything works the following message appears.



If this message does not display, an error message displays to help direct you on how to fix the issue. For example:

Dreamwea	ver MX 2004	×
<b>i</b>	Dreamweaver cannot use the prefix you entered to display live data. Please double-check your site configuration or click on Help for more information on how to correct this problem. (HTTP Error: 404)	
	Don't show me this message again.	
	ОК	

- 12. Click Next.
- 13. Choose if you want to copy your site to a remote server. For these example steps choose the **No** radio button.

When you are done editing a file, do you copy it to another machine? This might be the production web server or a staging server that you share with team members.



- 14. Click Next.
- 15. A summary appears.

Your site has the following settings:

Local info: (Local root folder will be created)

Site Name: MyFirstSite Local Root Folder: c:\inetpub\wwwroot\MyFirstSite\

Remote info:

Access: I'll set this up later.

#### Testing server:

Access: Local/Network Remote Folder: c:\inetpub\www.root\MyFirstSite\

Your site can be further configured using the Advanced Tab.

**NOTE** If you feel comfortable using Dreamweaver to create your Web site, you can click the Advanced Tab and further define your Web site settings. You can access help for the Advance Tab in Dreamweaver.

16. Click Done.

### Building the Site Map in the Files Window

Now that the site has been defined, the Files window appears on the right side of the Dreamweaver window. The Files window provides two visual representations of the local site: Site Map and Site Files.

The Files window can be used to add new files to the site, and to add, remove, and change links. It can also be used to create an

image file of the site that can be exported to and printed from an image editing application.



This is useful for creating a Web page with a visual layout of your site and linking each page to the image. This is also useful if you preparing a report on your site and would like to have a site map image included in the report.

To create a site map image, follow these steps:

- 1. In the **Files window**, click **Map view** from the dropdown box on the right.
- 2. Click the dropdown Menu as shown below.



- 3. Click File > Save Site Map.
- 4. In the File Type dropdown box, select .bmp or .png.
- 5. Enter a name in the File Name text box.
- 6. Select a location to save the file.
- 7. Click Save.

A home page needs to be created and defined in order to build the site map.



### Steps to Add a Home Page

- 1. Right click the local root folder in the Files window. For example, Site MyFirstSite (c:\inetpub\wwwroot\MyFirstSite).
- 2. Click New File.
- 3. A file titled **WebForm.aspx** appears in the Files window.

4. Right click WebForm.aspx and choose Edit > Rename.



5. Name the file Index.aspx.

6. Right click **Index.aspx** and select **Set as Home Page**.



7. To view a site map, click **Map view** from the dropdown menu in the Files window.



8. The Map view appears.



### **Creating a Subfolder and Files**

The home page currently has no links. After links have been established, the home page and the linked pages will be displayed in the site map.

### Steps to Create a Subfolder and Files

- 1. If you are in Map view, switch back to Local view.
- Right click the local root folder in the Files window. For example, Site - MyFirstSite (c:\inetpub\wwwroot\MyFirstSite).
- 3. Click New Folder.
- 4. Name the new folder Pages.
- 5. Create two files to put into the **Pages subfolder** by right clicking the subfolder and choosing **New File**.
- 6. Rename the two files Products.aspx and Services.aspx.

**NOTE** The template will be applied to these files in the following steps.

### **Creating the Template**

Templates allow a designer to construct a site that has a consistent style and is easy to update. Templates are convenient for a team of designers, allowing some members of the team to concentrate on the style of the pages, while other members of the team create new pages and revise the content without altering the formatted layout. Changes can be made to the overall design of the site quickly by revising the template files directly.

### Steps to Create a Template

The Create a Template example is broken into three parts:

- "Creating a New Template" on page 18
- "Creating a Navigation Bar for the Template" on page 19
- "Establishing Editable Regions and Saving the Template" on page 21

#### **Creating a New Template**

- 1. Click File > New...
- On the General Tab, click Template page > ASP.NET VB template.
- 3. Click Create.
- Choose Modify > Page Properties. In this dialog box you can do the following:
- choose a title for the template
- set the template's background
- set text
- choose links colors
- set left and top margin, and margin width and height.
- 5. Insert a two row, two column table in the template.
- The first row is reserved for the page banner and will be an editable region.
- The second row of the first column is where the navigation bar will be built. This row will be an uneditable region.

• The second row of the second column is where the future content will be placed.

It should look like this





#### Creating a Navigation Bar for the Template

Most Web sites have a navigation bar that provides links to the individual Web pages of the site. Usually, the navigation bar is inserted into the uneditable region of the template. This method ensures it is easily applied to each page created from the template and is updated by simply editing the template.

- 1. In the second row of the first column, insert a one column, seven row table.
- 2. In the first row of this embedded table, type **HOME**.
- 3. In the third row, type **PRODUCTS**.

- 4. In the fifth row, type **SERVICES**.
- 5. In the seventh row, type **CONTACT**.

Your page should look like this:

HOME	
PRODUCTS	
SERVICES	
CONTACT	

 With the cursor in a cell, the Property inspector displays the cell properties, such as background color (Bg), width (W), height (H), and text alignment (Horz/Vert). Apply the desired properties to each cell.

II ▼ Pro	perties				·				E,
<u>F</u> ormat	None	✓ St <u>y</u> le	None	✓ B I	EII	Link		<b>v</b>	?
Font	Default Fon	t 🔽 Size	one 🗸		≣ 5≣ 12 12	<u>T</u> arget	~		-
EF)	Cell	Horz Default	t 🔽 🖳	No wrap Bg			) 🗇 🗀 (	Page Properties	)
	i jį	Vert Default	t 🔽 <u>H</u>	Header 🗌 Bg		Brdr 🖵			

- 7. To link to index.aspx, products.aspx, and services.aspx, select the text in the cell, click the Folder icon in the Link field of the Property inspector, browse to the appropriate file, and select the file. Repeat this process for each cell.
- 8. For the Contact link, select the CONTACT text and, in the Link field of the Property inspector, type in **mailto:youremailaddress**. When a visitor to the site selects that link, their e-mail program opens and an e-mail with your address in the To: field becomes available.

### Establishing Editable Regions and Saving the Template

Once the design of the future template is complete, it must be saved as a DWT file (template) in order to be able to establish editable or noneditable regions.

Originally, each template has only locked regions. Locked regions can be edited only inside the template itself. Editable regions are placeholders for the content that is unique to each page to which the template is applied. The content of editable regions is created and edited in the document that is derived from the template.

- With the cursor in the first row, second column, select Modify > Templates > New Editable Region or click the Editable Region button (<sup>></sup>).
- 2. The New Editable Region dialog box appears.
- 3. In the **Name** field, type **Banner Editable Region** as the name for this region of the template. This editable region will be used as a placeholder for the title of the site pages.
- 4. Click **OK**. In the Document window, the region is highlighted and identifiable by the region name.



- With the cursor in the empty cell next to the navigation bar (the second row of the second column), Modify > Templates > New Editable Region or click the Editable Region button (2).
- 6. In the **Name** field, type **Body Editable Region** as the name for this region of the template. This editable region will be used as a placeholder for the text of the site pages.
- 7. Click **OK**. In the Document window, the region is highlighted and identifiable by the region name.



- 8. Click File > Save.
- 9. In the Save as: text box choose a name.

10. Click Save.

**NOTE** Make sure not to move the DWT file from the Templates folder. Also, do not rename or move the Templates folder. As long as the DWT file remains inside the Templates folder, the links from the pages derived from the template to the template will not be broken.

### Applying the Template

The new template is ready to apply to the index, products and services pages. In an applied template, the editable regions are highlighted. They appear in a highlight color on all pages to which the template is applied as well.

### Steps to Apply the Template

- 1. Select File > Open.
- 2. Browse to and select **Index.aspx**.
- 3. Select Modify > Templates > Apply Template to Page.
- 4. Click the **Template**.
- 5. Click **Select**.
- 6. In the table, type the title of the page in the **Banner Editable Region**. Type or insert text in the **Body Editable Region**.
- 7. Click File > Save.
- 8. Repeat these steps for the Products.aspx and Services.aspx pages.

### **Testing and Viewing the Site**

You are now ready to test and view the site.

#### Steps to Test and View the Site

- 1. In the Document window, select **File > Open**.
- 2. Browse to and select Index.aspx.
- 3. Select F12 to preview in the browser.
- 4. Click the navigation bar links to navigate from page to page.

# 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 .NET 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 25
- "Setting Up a Template" on page 26
- "Making the Server Controls Available" on page 27
- "Inserting Server Controls Using Drag and Drop" on page 29
- "Inserting Server Controls Programmatically" on page 33
- "Using Drag & Drop and Programmatically Together" on page 51
- "Data Binding with Server Controls" on page 53
- "Caching with Server Controls" on page 59
- "List of Server Controls" on page 61

Additional information and examples for Ektron Server Controls are available online by going to the Developers section of the sample site that was installed. Here is the path:

• http://localhost/CMS400Example/Developer.aspx

Replace localhost with the webroot where you installed the Example 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/CMS400Example/CMS400Example.sln.
- 2. The sample site project opens.
- 3. To work on a template page, click it from the Solution Explorer.

🌏 🖆 🖻	2 🛠 😅	
≡ ■	Ⅲ 征 律 律 •	
$\triangleleft \triangleright \times$	Solution Explorer - CMS400Intranet 🛛 🗛 $ imes$	
_		
	🗈 🖷 🧱 article.aspx 📃	
	AssemblyInfo.vb	
	🕀 📄 BobsTest.aspx	
	🕀 😇 calendar.aspx	
_	🕀 🐨 🚾 corporate.aspx	
	- Al default.css	l
	- \Lambda DHTML Menu Styles.css	
	- A ekcalendar.css	
	🖅 🗒 employee, information, aspx	
	E kr.asnx	
	E br forms aspy	
	Solution Explo 🖾 Class View 🎣 Search	
	Properties 4 ×	
	corporate.aspx File Properties	]
	Advanced	

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.

			Solution Explorer 🛛 💁 Cl	ass View  🧔 Searc	h	
		Pr	operties		Р	Х
		Fe	ormBlock1 ServerControl	Library.FormBlock		•
		•				
_			CssClass			
		Ð	Font			
			ForeColor			
			Behavior			
			AccessKey			
	-		Enabled	True		
	•		EnableViewState	True		
			TabIndex	0		
			ToolTip			
			Visible	True		
д	Х	] 0	Data			
			(DataBindings)			
	Π	10	Ektron	)		
			Authenticated	False		
			DefaultFormID	0		
	_	D	efaultFormID			
	-	TH	ne default ID of the Form Blo	ck to be displayed.		

# 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.

Add New Item - CM5400Intranet				×
Categories:	<u>T</u> emplates:			6-6- 6-6- 6-6-
Web Project Items UI Code Data	Web Form	Web Service	Mobile Web Form	•
Utility	Class	Module	Component Class	
	Data Form Wizard	DataSet	Web User Control	•
A form for Web Applications				
Name: WebForm3.aspx				
	Open	Cancel	Help	

3. Add controls to determine the page content.

# Making the Server Controls Available

Νοτε

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 .NET toolbox (View > Toolbox).
- 2. Right click the mouse within the Toolbox.
- 3. Click Add Tab.

4. Type Ektron Server Controls then press enter.

Components	<ul><li>▼</li></ul>
HTML	
Clipboard Ring	
General	
Ektron Server Conti	rols
Server Expl	* Toolbox
Output	

- 5. Click the Ektron Server Controls Tab.
- 6. Right click the mouse in the empty area.
- 7. Click Add/Remove Items.
- 8. The Customize Toolbox dialog appears.

Name	Namespace	Assembly Name I
ADODC	Microsoft.VisualBasic.Comp	Microsoft.VisualBasic.Compatibility.Dat
ADODCArray	Microsoft.VisualBasic.Comp	Microsoft.VisualBasic.Compatibility.Dat •
🗹 AdRotator	System.Web.UI.MobileCont	System.Web.Mobile (1.0.5000.0)
🗹 AdRotator	System.Web.UI.WebControls	System.Web (1.0.5000.0)
AssemblyInstaller	System.Configuration.Install	System.Configuration.Install (1.0.5000 )
🗹 Button	System.Windows.Forms	System.CF.Windows.Forms (7.0.5000.0)
🗹 Button	System.Web.UI.WebControls	System.Web (1.0.5000.0)
Button	System.Windows.Forms	System.Windows.Forms (1.0.5000.0)
ButtonArray	Microsoft.VisualBasic.Comp	Microsoft.VisualBasic.Compatibility (7.0 )
🗹 Calendar	ServerControlLibrary	ServerControlLibrary (1.0.1787.20128) 👘
√	e i ortigende i	
ADODC		Browse
🤬 Language: Ir	nvariant Language (Invariant Count	ry)

- 9. Select the .NET Framework Components tab.
- 10. Browse to the directory that stores Ektron CMS400.NET's dll files, localhost/CMS400Example/bin.

11. Add the Ektron.Cms.Controls.dll file and press **OK**. This file provides access to **Ektron CMS400.NET**'s server controls.

### **Removing the Server Controls**

- 1. Display the Visual Studio .NET toolbox (View > Toolbox).
- 2. Right click the mouse within the Toolbox.
- 3. Click Add/Remove 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 29

For information on adding server controls, see "Making the Server Controls Available" on page 27

# Inserting Server Controls Using Drag and Drop

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



- 3. Drag a server control and drop it into the desired location on the page.
- 4. As desired, modify the control's properties using the Properties area of Visual Studio .NET.

		Solution Explorer 🛛 💁 Cla	ass View   🧔 Search		
	Pr	operties		<b>ņ</b>	×
	E	ormBlock1 ServerControl	Library.FormBlock		•
- 11		CssClass			
	Ð	Font			
		ForeColor			
		Behavior			
		AccessKey			
		Enabled	True		
-		EnableViewState	True		
		TabIndex	0		
		ToolTip			
		Visible	True		
<b>中</b> ×	] 8	Data			
<b></b>		(DataBindings)			
	10	Ektron			
_		Authenticated	False		
		DefaultFormID	0		-
-		efaultFormID ne default ID of the Form Blo	ck to be displayed.		

For a description of server control properties, see "List of Server Controls" on page 61.

### 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 /CMS400Example/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/cms400example/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 and content blocks. For example, if you insert a List Summary Server Control, its **FolderPath** property identifies the folder whose contents are displayed. If you are not sure of the folder path, click the gray button.

Direction	Ascending	
DisplayXslt	ecmNavigation	
FolderPath	۱.	
Hide	False	
Language	1033	

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 [CBR R31]. 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.

CMS Login	
Status: Log	ged Out
Username:	
Password:	
Domain:	http://localhost/CMS400Samp Select
	Close Login Logout

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.

Ektron CMS Explorer					
B-Folders	Language:	All		Folder Sel	ection
⊞ Human_Resources	Title		ID	Description	Path
⊞-Marketing -MIS/IT -News -Sales ⊞-Users					
Select Cancel					

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 .NET, 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 .NET 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 29), 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 51.

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

- "Steps to Insert a Server Control Programmatically" on page 34
- "Server Control Properties in Code Behind" on page 38
- "Customizing Server Controls" on page 48

### **Steps to Insert a Server Control Programmatically**

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

- "Step 1: Declare the Namespace" on page 35
- "Step 2: Create an Instance of the New Control" on page 35
- "Step 3: Set the Properties" on page 36
- "Step 4: Call the Fill Method" on page 37

"Step 5: Use .text Property to Display the Control" on page 37

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.

(Page	> Events)
Impo	orts Ektron.Cms.Controls
🗆 Publ	lic Class WebForm1

#### Νοτε

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 61. 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 38.

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 = 1

or

dim MyColl as new Ektron.Cms.Controls.Collection

MyColl.DefaultCollectionID = 1

#### With C#, use this syntax.

Ektron.Cms.Controls.Collection MyColl = new

Ektron.Cms.Controls.Collection();

MyColl.DefaultCollectionID = 1;

This line tells the page to display CollectionID 1 unless otherwise specified.

**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 32.) 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 = 1

MyColl.Random = True

This line sets the Random property to **true**. See Also: "Random" on page 116.

### 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 = 1
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 = 1
MyColl.Random = True
MyColl.Fill()
Labell.Text = myColl.EkItems(0).Title
```

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

```
Dim myColl As New Collection
Dim ekitem As New Ektron.Cms.Common.ContentBase
myColl.DefaultCollectionID = 2
myColl.Fill()
Labell.Text = ""
For Each ekitem In myColl.EkItems
Labell.Text &= "<a href=""" & ekitem.QuickLink & """>" & ekitem.Title & "</a>"
Next
Labell.Text &= ""
```

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
- labell.Text &= "a href=" & ekitem.QuickLink & ">" & ekitem.Title & "</a>
   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 38
- "Mutual Server Control Properties in Code Behind" on page 39
- "Accessing Additional Properties" on page 41
- "Accessing Items in an Array" on page 47
- "Referencing the Page Property" on page 47

### **Displaying Properties for a Control**

You can use Visual Studio .NET'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 61. To learn about native VS .NET 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 50.

### 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
lsLoggedIn	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
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.

```
Literall.Text = ContentBlockl.IsLoggedIn
If ContentBlockl.IsLoggedIn Then
Literal2.Text = ContentBlockl.loggedInUserName & " is logged in "
Literal3.Text = ContentBlockl.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.

WebForm5 - Microsoft Internet Ex					
<u> </u>	<u>E</u> dit	<u>V</u> iew	F <u>a</u> vorites	<u>T</u> ools	Ŀ
G	Back	• 6	) - 💌	2	6
Addres	ss 🙆	http://k	calhost/CM	S400Exa	ampl
Links	ど Ekt	ronNet	Welcome	🛃 Ektr	on, 1
True					
John Edit is logged in					
2 is the User ID					

#### 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

label1.text = MyCB.EkItem.dateCreated

or

dim MyCB as new Ektron.Cms.Controls.ContentBlock

label1.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 47.

With C#, use this syntax.

Ektron.Cms.Controls.ContentBlock MyCB = new

Ektron.Cms.Controls.ContentBlock();

label1.text = MyCB.EkItem.dateCreated

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	
ContentStatus	<ul> <li>The status of the content block.</li> <li>approved</li> <li>checked out</li> <li>checked in</li> <li>expired</li> <li>pending deletion</li> <li>pending expiration</li> <li>pending start date</li> <li>submitted</li> </ul>	Ektron CMS400.NET User Manual, "Appendix A: Content Statuses."
ContentType	One of the following <ul> <li>all types</li> <li>archived content</li> <li>archived forms</li> <li>content</li> <li>forms</li> </ul>	To learn about archived content, see the <b>Ektron CMS400.NET</b> 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.	

Property	Description	For more information, see
DisplayDateModified	The date when the content block was edited. It is formatted as a string that represents Ektron CMS400's display of the date.	
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> <li>archive display</li> <li>archive expire</li> <li>refresh report</li> </ul>	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</a>	

Property	Description	For more information, see
ld	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"
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	The status of the content block <ul> <li>approved</li> <li>checked out</li> <li>checked in</li> <li>expired</li> <li>pending deletion</li> <li>pending expiration</li> <li>pending start date</li> <li>submitted</li> </ul>	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. <b>Ektron CMS400.NET</b> only uses Xslt1, 2 and 3 in the workarea.	
Xslt5	The developer can use this property programmatically. <b>Ektron CMS400.NET</b> 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. *Ekltems* is an array of Ektron.Cms.Common.ContentBase. *Ekltem* is a single Ektron.Cms.Common.ContentBase.

Search, Collection and ListSummary have Ekltems (an array of ContentBase), while and ContentBlock have a single Ekltem. Here is an example of how to use this feature.

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

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

#### Referencing the Page Property

If a control requires a reference to its 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 = me
MySearch.Fill()
With C#, use this syntax.

Ektron.Cms.Controls.Search MySearch = new Ektron.Cms.Controls.Search();

```
MySearch.Page = this;
```

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 .NET'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.

Νοτε

If you do not see the code behind files, click the Show All Files button on the VS .NET 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 49
- "Recognizing the Server Control Within the Code Behind" on page 49
- "Customizing the Server Control in the Code Behind" on page 50

• "Troubleshooting Error Creating Control Message" on page 50

#### Recognizing the Server Control Within the HTML

Within a Web page's HTML, a <cms> tag wraps the VS. NET 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 .NET code behind file, the Ektron server controls appear (along with the VS .NET 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.

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 61.

#### 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.

Literal "DhtmlJavaScript"]:	🔁 🐻 🕞 🛙
RC International	Solution 'C
ноте   [ Literal "DropDownMenu" ]	
Support News Careers Search Calendar	• E
······	
Error Creating Control - ctrlTopContentBlock 0	
Error Creating Control - ctrlMainContentBlock  Parser Error: File or assembly name Ektron.Cm or one of its dependencies, was not found.	is.Common,
Error Creating Control - ctrlBottomLeftContentBlock 🔨 Error Creating Control - ctrlBottc	
	🛛 🐼 s 💁 c 🔍
	Properties
	Ektron.Cms.E
Copyright 2003 Ektron, Inc. Sample Site Site Languag	

## 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.

Toolbox	ф X	Start Page	Intranet\WebFo	orm5.aspx *	Intranet\WebForm5.a	sp
Data						
Web Forms		ASP.NE	Unleashed			
Components		Content Management Bible				
HTML		HTML for the World Wide Web with XHTML and CSS				
Clipboard Ring		Microsoft	ASP.NET Pro	gramming w	ith Microsoft Visu	al
General		<u>C#.NET</u>				
Ektron Server Controls		Simply Vi	isual Basic.NET	2003		
Pointer		Visual Ba	sic.NET How to	o Program		
🔟 Calendar		Propert	ies		×	1
Collection		LintCu	mmamul Ektern	Controle I	iatCummon 🖉	
ContentBlock			Innary1 Ektron.c	uns.controis.c		4
FormBlock		2 E E	, 🔲 🖾			
E LanguageAPI		🗄 🗉 Ektr	on		<u> </u>	
LanguageSelect		:: Auth	enticated	True		
E ListSummary		Cont	entParameter			ŀ
		Direc	tion	Ascendi	ng	F
		Disp	layXslt	ecmNavig	ation	F
Renu		Fold	erPath	\Books		E
🥨 MetaData		Hide		False		ŀ
MetaDataList		: Lang	uage	1033		
A Search		:: MaxF	Results	0		1
		:: Orde	rKey	Title		
		:: Rand	lom	False		
		🔅 Recu	rsive	False	-	

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=""" & ListSummaryl.EkItems(i).QuickLink & """>"
& ListSummaryl.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.

/iocainosc/cms+ooexampie/inua			
Browse - WebForm5			
1/4/2005 4:23:36 PM			
1/4/2005 7:35:38 PM			
1/4/2005 7:31:25 PM			
1/4/2005 4:27:47 PM			
1/4/2005 7:26:08 PM			
1/4/2005 4:17:18 PM			

### **Data Binding with Server Controls**

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

Title	ID Teaser
Plastic Molder #123	13 RC International is looking for an experienced plastics molder.
Servo Control Engineer #124	14 RC International is looked for a highly skilled servo control eng

The following are data bindable Ektron Server Controls.

- Collection
- Content Block
- Form Block
- List Summary
- Search

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

```
Dim myCol As New Ektron.Cms.Controls.Collection
myCol.DefaultCollectionID = 1
myCol.Fill()
DataGridl.DataSource = myCol
DataGridl.DataBind()
```

**NOTE** When using drag and drop server controls for the Collection and DataGrid, you need to call the DataBind() method in the code behind. An example line of code would be: DataGrid1.DataBind().

This subsection contains the following topics:

- "Steps to Data Binding Using Drag and Drop Server Controls" on page 54
- "Customizing the DataGrid Table View" on page 57

#### Steps to Data Binding Using Drag and Drop Server Controls

The following example uses a Collection.

1. Drag and drop a data bindable Server Control on the Web form.



2. In properties, choose the Ektron default ID for the Server Control and make sure the XSLT is set to **None- DataBind Only**.

C	Collection1 Ektron.Cms.Controls.Collection				
•					
Ξ	Ektron				
	Authenticated	True			
	CmsSitePath	http://localhost/CMS400			
	DefaultCollectionID	1			
	DynamicParameter				
	GetHtml	False			
	Language	1033			
	Random	True			
	Xslt	None - Databind Only			

3. Drag and drop a DataGrid on the web form.

			•	· · ·	
			-	Ŧ	
<b>4 X</b>		Start Page	WebForm	1.aspx *	WebFo
		∎ Collectio	>n[ID=1]	- Count: 1	1
		n Column0	□ Column1	Column2	
r i		abc	abc	abc	
:on		abc	abc	abc	
	-	abc	abc	abc	
List		abc	abc	abc	
		abc	abc	abc	

4. In the DataGrid properties, choose the DataSource.

Pr	operties		ą.	x
DataGrid1 System.Web.UI.WebControls.DataGrid				•
•	. <u>₽</u>			
Ξ	Data			
	(DataBindings)			
	DataKeyField			
	DataMember			
	DataSource 🛛 🕲	Collection1	-	

- 5. The DataGrid is populated with column headers and the type of information that will be in the table.
- 6. Go to code behind and add DataGrid1.DataBind() to the Page\_Load event.

```
Private Sub Page_Load(ByVal sender As System.Object
    'Put user code to initialize the page here
    DataGrid1.DataBind()
End Sub
```

LastEditorFname	DateModified	Html	QuickLink	HasErrors	Teaser	D
					RC Internationals most successful entry in to the	
Application	2/17/2004 4:12:11 PM		/CMS400Sample/products.aspx? id=5	False	competitive RC racing world is the RC Cheetah.	5

## 7. Switch to Design mode and right-click on the DataGrid. Then, choose **View in Browser**. The table appears.

#### **Customizing the DataGrid Table View**

Customizing the DataGrid is a great way to display and format specific information from a data source. You can select formatting for columns, paging and borders, plus other general formatting for the table. In the following example, we choose table columns. 1. In Web form design, right click the DataGrid and choose **Property Builder**.



2. Click **Columns** and uncheck the **Create columns automatically at run time** checkbox.

DataGrid1 Proper	ties	
General	Column list Available columns:	run time
A <u>A</u> Format ∰ Borders	Column properties	* * * *

3. Choose the fields for the custom table from the available columns.

DataGrid1 Prope	erties	
General Columns Paging A <u>A</u> Format Borders	Create columns automatically at run time         Column list         Available columns:         Selected columns:         Image: Column list         Available columns:         Image: Column list         Image: Column list <t< th=""><th>* * X</th></t<>	* * X
	BoundColumn properties	

- 4. Click **OK**.
- 5. Right click the DataGrid and choose View in Browser.
- 6. The custom table appears.

Title	QuickLink	D
RC Lilly	/CMS400Sample/products.aspx?id=6	6

For more information on working with DataBind() or DataGrid, review the help topics in Visual Studio.

### **Caching with Server Controls**

One of the most important factors in building high-performance, scalable Web applications is the ability to store items in memory the first time they are requested. These items include data objects, pages, or parts of a page.

Known as caching, this process allows you to use a number of techniques to store page output or application data across HTTP requests and reuse it. You can store items on the Web server or other software in the request stream, such as the proxy server or browser.

With caching, the server does not have to recreate information, particularly information that demands significant processor time or other resources. This saves valuable time and resources.

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, further define if the time is cached, based on the user's status.

- 1. Create a new Web form in your Ektron CMS400.NET project.
- 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.
- Set the DefaultContentID. For this example, DefaultContentID = 20.
- 6. Add the following code 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 with your browser.

9. Press the refresh button on your browser.

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

## List of Server Controls

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

Server Control Names	Description	More Information
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 197
Calendar	Displays an event calendar.	"Calendar Server Control" on page 126
Collection	Displays links generated by collections.	"Collection Server Control" on page 119
ContentBlock	Displays and manages a content block.	"ContentBlock Server Control" on page 70
DhtmlMenu	Displays a drop down menu with submenus	"DhtmlMenu Server Control" on page 177
FormBlock	Displays a content block that is associated with a form.	"FormBlock Server Control" on page 80
IndexSearch	Perform XML index searches on XML content.	"IndexSearch Server Control" on page 96
		For information on using XHTML Search, see "Search Server Control" on page 84
LanguageAPI	Allows developers to force a language for a Web site.	"LanguageAPI Server Control" on page 67
LanguageSelect	Lets user select language of site.	"LanguageSelect Server Control" on page 65

Server Control Names	Description	More Information
ListSummary	Displays a list of content block summaries.	"ListSummary Server Control" on page 113
Login	Displays the login or logout button, which lets the user log in and use Ektron CMS400.NET.	"Login Server Control" on page 63
Menu	Calls a defined menu to be used with an Xslt to display the menu.	"Menu Server Control" on page 160
MetaData	Adds metadata contained in content blocks to a Web page.	"MetaData Server Control" on page 185
MetaDataList	Creates a list of content blocks based on the metadata contained in the content blocks.	"MetaDataList Server Control" on page 188
RssAggregator	Aggregates an RSS feed from a RSS Web site.	"RssAggregator Server Control" on page 193
Search Inserts a text box with a search button in the template. The site visitor enters criteria and launches a search of the XHTML content on your Web site.		"Search Server Control" on page 84
		For information on using XML Indexing search, see "IndexSearch Server Control" on page 96

# 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 <b>Ektron</b> <b>CMS400.NET</b> User Manual section "Logging In and Out" > "Site Preview."
<b>X</b> Preview	Turns off site preview mode.

This subsection contains the following topics:

- "Placing a Login Button" on page 63
- "Login Server Control Properties" on page 64

### 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.Net 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 32	String
Hide	Used to hide login button in design time and run time. <b>True</b> = Hide login button <b>False</b> = 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
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	String

# LanguageSelect Server Control

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

This control lets a visitor to your Web site 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 right corner of the home page.

This subsection contains the following topics:

- "LanguageSelect Server Control vs. LanguageAPI Server Control" on page 65
- "LanguageSelect Server Control Properties" on page 66

### 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 67.

## 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.Net 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 32	String
Hide	Used to hide language select dropdown box and label in design time and run time. <b>True</b> = Hide language select dropdown box and label <b>False</b> = Show language select dropdown box and label	Boolean
LabelName	Lets you define the label next to the language select dropdown box. The default label is <b>Language</b>	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
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	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 browsers language, and display the site in that language.

This subsection contains the following topics:

- "LanguageSelect Server Control vs. LanguageAPI Server Control" on page 67
- "LanguageAPI Server Control Properties" on page 67
- "Code Behind Only Properties" on page 68

### 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 65.

#### 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.Net 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**

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

Property	Value	Data Type
DefaultLanguage	<b>Read only</b> . What the default language of the site is set to. For example, the demo site is "English (Standard)".	String
DefaultLanguageID	<b>Read only</b> . 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

Property	Value	Data Type
LanguageIdList	Read only. Lists all the language IDs that are activated in Ektron CMS400.NET.	Array of Integers
	For information on how to enable languages, see The Administrator Manual > "Multi-Language Support" > "Working with Multi-Language Content" > "Determining Which Languages are Available".	
LanguageTitleList	<b>Read only</b> . Lists all the languages that are activated in <b>Ektron CMS400.NET</b> .	Array of Strings
MultiLanguageEnabled	<b>Read only</b> . Tells if the site supports multi-language mode.	Boolean
	True = Multi-language enabled	
	False = Multi-language not enabled	
	Multi-language is enable by default in <b>Ektron</b> <b>CMS400.NET</b> . To disable Multi-language, see Administrator Manual > "Multi-Language Support" > "Working with Multi-Language Content" > "Disabling Support for Multiple Language Content".	

# **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 78

This subsection contains the following topics:

- "Static Content Block" on page 70
- "ContentBlock Server Control Properties" on page 71
- "Dynamic Content Block" on page 72
- "XML Content Block" on page 75
- "The OverrideXslt and DisplayXslt Properties" on page 76
- "Using the ContentBlock Server Control Programmatically Example" on page 78

### 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.Net 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 32	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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer
DynamicParameter	Select <b>None - Use Default</b> . This parameter is used for dynamic content blocks.	String
Hide	Used to hide a content block in design time and run time. <b>True</b> = Hide content block <b>False</b> = 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
OverrideXslt	Ignore for a non-XML content block.	must be one of these values: default, one, two, three
DisplayXslt	Ignore for a non-XML content block.	String

Property	Value	Data Type
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.</span>	String
	<ul> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	

Ξ	Ektron		
	Authenticated	True	
	DefaultContentID	28	
	DisplayXslt		
	DynamicParameter		
	Hide	False	
	Language	1033	
	OverrideXslt	Default	

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.

😫 News: RC Intern	ational Announces the RC Cheetah - Microsoft Internet Explorer
File Edit View	Favorites Tools Help
🖨 Back 🔹 🔿 🛬	🗿 🔄 🚮 😡 Search 🔝 Favorites 🐠 History 🛛 🛃 🗃
Address http://	www.YourSite.com/CMS400/news.aspx?id=11
RC III	ternational
Home   Products	;   Support   <b>News</b>   Careers   Search   Calendar

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.Net 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 32	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 32	Integer
DynamicParameter	Select <b>id</b> . When you do, this server control uses the content block passed as a URL parameter.	String

Property	Value	Data Type
Hide	Used to hide a content block in design time and run time. <b>True</b> = Hide content block <b>False</b> = 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
OverrideXslt	Ignore for a non-XML content block.	must be one of these values: default, one, two, three
DisplayXslt	Ignore for a non-XML content block.	String
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	String

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

Ξ	Ektron	
	Authenticated	False
	DefaultContentID	1
	DisplayXslt	
	DynamicParameter	id 🔹
	Hide	False
	Language	1033
	OverrideXslt	Default

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.Net 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 32	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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer
DynamicParameter	To make this content block dynamic, select <b>id</b> . 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. <b>True</b> = Hide content block <b>False</b> = 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
OverrideXslt	An internal Xslt file used to display XML content if one is not identified in the Xslt property. See Also: "The OverrideXslt and DisplayXslt Properties" on page 76	must be one of these values: default, one, two, three
DisplayXslt	Specify an external XSLT file. See Also: "The OverrideXslt and DisplayXslt Properties" on page 76	String

Property	Value	Data Type
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	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 XML 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
The default XSLT specified in the Edit XML Configuration screen (illustrated below). Note that, in this example, the XSLT Packaged option is the default XSLT, since it is selected. XSLT Packaged is the XSLT from the Edit XML configuration screen (that is, the XSLT created in the Data Designer).  Edit XML Configuration "new configuration"      ←		Default
Title: new configuration   ID: 6   Description: xslt 2   Display Information: (Files prefixed with/CMS300SampleTest/XmlFiles   Default:   0 XSLT 1:   0 XSLT 2:   0 XSLT 3:   0 XSLT 9:   0 XSLT Packaged:		
XSLT 1 from the Edit XML configuration screen		1
XSLT 2 from the Edit XML configuration screen		2
XSLT 3 from the Edit XML 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 XML configuration screen. For example sample.xslt	does not matter - if XSLT value exists, <b>OverrideXSLT</b> is ignored

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

## 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<br/>Web form.

 Dim MyContentBlock As New ContentBlock

 MyContentBlock.DefaultContentID = 8

 MyContentBlock.Fill()

 Literall.Text = MyContentBlock.EkItem.Html

 To display a content block with the content block title, do the<br/>following:

 NOTE

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

 Dim MyContentBlock As New ContentBlock

 MyContentBlock.DefaultContentID = 8

 MyContentBlock.Fill()

Literall.Text = MyContentBlock.EkItem.Title Literal2.Text = MyContentBlock.EkItem.Html

# **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?	O Floating Holiday O Vacation O Sick Leave O FMLA O Other:		
Dates of Absence: Total Days Absent:	mm/dd/yyyy O With Pay O Without Pay	to mm/dd/yyyy	
Comments:		×	

This subsection contains the following topics:

- "FormBlock Server Control Properties" on page 81
- "Automatic versus Manual Generation of Form Tags" on page 83

## FormBlock Server Control Properties

The following table explains the properties of the FormBlock Server Control.

#### Νοτε

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.Net help.

Property	Value	Data Type
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. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32.	String
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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32.	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
FormName	Set this value to the name of your main runat=server form.	String
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: "Automatic versus Manual Generation of Form Tags" on page 83	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
Property	Value	Data Type
----------	---	-----------
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.</span>	String
	<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.</div>	

### 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 <b>True</b> .
Modify HTML form tags in the .aspx file	An .aspx page and associated code- behind web form	Set Include Tags property to <b>False</b> . Here is the default .NET generated form tag: <form <br="" id="Form1" method="post">runat="server"&gt; Modify the form tag as indicated in red: <form <br="" id="Form1" method="post">runat="server" OnSubmit="EkFmValidate(this);"&gt;</form></form>

## **Search Server Control**

The Search Server Control displays a text box with a search button on the template. To learn about which content is considered by this search function, see the **Ektron CMS400.NET** Administrator manual section "Searching Content on the Web Site."

When added to a template and visited, the Search Server Control looks like this. Users can enter text and click the search button to search for XHTML content on the Web site.

Search
$oldsymbol{eta}$ All the words
${f C}$ Any of the words
C Exact Phrase
🗖 Match Partial Words

This subsection contains the following topics:

- "Search Server Control vs. IndexSearch Server Control" on page 85
- "Search Server Control Properties" on page 85
- "Search Server Control Property Example" on page 88
- "Text Box Examples" on page 89
- "Button Image Source Examples" on page 90
- "Search Button Text Examples" on page 91
- "Display Examples" on page 92
- "ShowDate Examples" on page 92
- "Specify Search Control To Display Search Results on a Separate Page" on page 93

# Search Server Control vs. IndexSearch Server Control

The Search Server Control lets users search your Web site for XHTML Content. 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 96.

#### Search Server Control Properties

The following table explains the properties of the Search 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.Net 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>FolderPath</b> property.	String	"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32
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 90
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 91
Display	Determines if the search options are in a row ( <b>Horizontal</b> ) or a column ( <b>Vertical</b> ).	either vertical or horizontal	"Display Examples" on page 92

Property	Description	Data Type	For more information, see
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	ecmNavigation - "Example of ecmNavigation Display" on page 121 ecmTeaser - "Example of ecmTeaser Display" on page 122
EmptyResultMsg	The return message displayed, if the search returns no hits. The default is: Search returned zero results	String	
FolderPath	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	"Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32
Hide	Used to hide Search Server Control in design time and runtime. True = Hide control False = Show control	Boolean	
Language	Set a language for viewing search content. The language property shows results in design-time (in Visual Studio) and at run-time (in a browser).	Integer	
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).	Integer	

Property	Description	Data Type	For more information, see
OrderDirection	The direction the search results will be sorted in. The default is <b>Ascending</b>	String	
	• Ascending - Results from A to Z		
	Descending - Results from Z to A		
PostResultsAt	Enter the Web form where the search results will be posted. When left blank, the results are posted to the Web form where the search was initiated.	String	"Specify Search Control To Display Search Results on a Separate Page" on page 93
	<b>Important:</b> A search control must exist on the posting page.		
Recursive	Whether to search sub-folders of the identified root folder. The root folder is identified in the Folderpath property.	Boolean	
ShowDate	If <b>True</b> , displays the content block's last modified date. This is useful for determining the most recent content block.	Boolean	"ShowDate Examples" on page 92
ShowSearchBoxAlways	If set to <b>false</b> , the search box does not appear on PostBack	Boolean	
Text Box Size	The size of the display text box for user input. The width is based on the number of characters.	Integer	"Text Box Examples" on page 89

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

#### Search Server Control Property Example

The following example places a text field and a search button on a template.

Ektron	
Authenticated	True
ButtonImgSrc	
ButtonText	Search
DefaultResultDisplay	True
Display	Vertical
DisplayXslt	ecmNavigation
FolderPath	\Marketing
Hide	False
Language	1033
MaxCharacters	50
MaxResults	0
Recursive	True
ShowDate	True
ShowSearchBoxAlways	True
TextBoxSize	25
	Ektron Authenticated ButtonImgSrc ButtonText DefaultResultDisplay Display DisplayXslt FolderPath Hide Language MaxCharacters MaxResults Recursive ShowDate ShowSearchBoxAlways TextBoxSize

The above example has the following parameters:

• No image is used for the button

- The text on the submit button is Search
- The results will display on the page
- The search options are arranged vertically
- Searches content in the \content\marketing folder
- The search allows a maximum of 50 characters
- Returns unlimited results when the search is performed
- Search is *recursive*
- Shows the last modified date of the content
- The search box will appear on PostBack
- The size of the text box is **25** characters

#### **Text Box Examples**

Value	Web Page View
10	Search © All the words © Any of the words © Exact Phrase □ Match Partial Words
50	Search All the words Any of the words Exact Phrase Match Partial Words

#### **Button Image Source Examples**

Value	Web Page View
(blank)	Search All the words Any of the words Exact Phrase Match Partial Words
search.jpg	Search All the words Any of the words Exact Phrase Match Partial Words
http://www.ektron.com/images/search.jpg	Search All the words Any of the words Exact Phrase Match Partial Words

#### Search Button Text Examples

Value	Web Page View
"Click Me to Search"	Click Me to Search Click Me to Search All the words Any of the words Exact Phrase Match Partial Words
"Click Me to Search"	<ul> <li>Search</li> <li>○ All Click Me to Search</li> <li>○ Any of the words</li> <li>○ Exact Phrase</li> <li>□ Match Partial Words</li> </ul>

#### **Display Examples**

Value	Web Page View
Vertical	
	Search
	All the words
	C Any of the words
	C Exact Phrase
	Match Partial Words
Horizontal	
	Search C All the words C Any of the words C Exact Phrase C Match Partial Words

#### ShowDate Examples

Value	Web Page View
False	
	Search Results
	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

Value	Web Page View
True	
	Search Results
	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)

# Specify Search Control To Display 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 have the search dialog box in a navigation pane, and the results displayed on a separate Web form.

1. Drag and drop a Search Server Control on a Web form.

WebForm.aspx	
P	,
Search	
• All the words	
O Any of the words	
C Exact Phrase	
Match Partial Words	

2. Set the following Ektron properties for the Search Server Control.

**DefaultResultsDisplay** = False

**DisplayXslt** = None - Databind Only

**PostResultAt** = The Web form name where the results are posted. For example, Webform2.aspx

**NOTE** webform2.aspx represents the name of the second Web form.

- 3. Add a second Web form to the project.
- 4. Drag and drop a Search Server Control on it.

WebForm2.aspx		
P		
Search		
• All the words		
○ Any of the words		
© Exact Phrase		
C Match Partial Words		

- 5. Set the Ektron property **ShowSearchBoxAlways** to false.
- 6. Build the solution.
- 7. In your browser, go to the first Web form.

😈 Back 🔹 💟 🐘 본	🖻 🚺	Search 7
Address Address Address Address Address	MS400Example/we	bform.aspx
Links ど EktronNet Welcome	🕖 Ektron, Inc.	CMS400 E
Search <ul> <li>All the words</li> <li>Any of the words</li> <li>Exact Phrase</li> <li>Match Partial Words</li> </ul>	]	

8. Use the search dialog.



9. The search results are loaded and displayed in the second Web form automatically.

🕻 🌍 Back 🔹 🜍 🖈 📓 🌠 🎾 Search 🎇 Favorites 🔣 🔯 🦃 🛀 📘
Address an http://localhost/CMS400Example/webform2.aspx
Links 🝘 EktronNet Welcome 💰 Ektron, Inc. 💩 CMS400 Example 🚳 Welcome to RC International 💩 We
Web Server         eWebEditPro+XML V4.2 supports the vision of create content once, reuse many times         Ektron, Inc. Named One of New England's Fastest Growing Technology Companies         Ektron Supports Rapid and Efficient Globalization Strategies on the Web         Ektron Offers a Visual Development Environment for Rapid CMS Integration and Deployment         Ektron Introduces an Enhanced Workflow Suite         Ektron Expands Presence with Marketing, Interactive and Web Design Firms         Ektron Announces Winner of "All-Stars" Customer Competition

Corporate

CMS Developer ....

## IndexSearch Server Control

Use the IndexSearch Server Control to display an XML configuration'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.

Genre	novel 👻			
Price	Between	<mark>∼</mark> 1		
Last Name	Jones 💌			
Paperback				
Publishdate	Select	~		
Title	Contains	👻 cms		
	(	Submit Sear	ch	

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 97
- "IndexSearch Server Control Properties" on page 97
- "SearchParmXML Property" on page 102
- "Using the IndexSearch Server Control Programmatically Example" on page 105

- "Programmatically Predefined General Search Results Replaced By Specific Search Results." on page 108
- "Turn Off Caching In Netscape Navigator and Mozilla Firefox." on page 112

#### IndexSearch Server Control vs. Search Server Control

See, "Search Server Control vs. IndexSearch Server Control" on page 85.

#### 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.Net 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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	String	
ButtonText	The text that appears on the button.	String	The default is: Search

Property	Description	Data Type	Values
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.
DisplayXslt	Determines how the information on the page is displayed	String	None-databind only ecmNavigation - lists the title of every content block in the folder See Also: "Example of ecmNavigation Display" on page 121 ecmTeaser - lists the title of every content block in the folder plus the content summary See Also: "Example of ecmTeaser Display" on page 122 Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page
EmptyResultMsg	The message that displays if the search returns no hits.	String	The default value is: Your search did not match any documents.
Folderld	The numeric id of the folder that you want to search. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32 The Recursive property determines whether the IndexSearch Server Control will use this folder's child folders in the search.	Integer	Any folder's numeric ID <b>0</b> (zero) represents the root folder.

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	True = Hide results False = show results
HitsMarker	This symbol shows how many search criteria each hit matched. The following example shows the first hit met three of the criteria. The second group met one criterion.	String	* # + (Custom) Custom lets you choose the marker to use. The custom marker can be any tex numeric, or symbol character except & and <.
	****         Microsoft ASP.NET Progr         *         Content Management Bible         Visual Basic.NET How to I         ASP.NET Unleashed         Simply Visual Basic.NET 2         This property is used when         the weighted property is set to         true. See Also: "Weighted" on         page 102.		For more Information, see The Ektron CMS400.NET Administrator Manual section "XML Indexing" > "Displaying Search Results"

Property	Description	Data Type	Values
Labelafter Labelbefore LabelBetween LabelContains LabelContains LabelContains LabelEqual LabelEqual LabelExactPhrase LabelGreaterThan LabelGreaterThan Labelinputdate Labelinputnumber LabelInputnumber LabelLessThan LabelOn LabelSelect	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> .	String	Examples of what the user will see on the site in French and English.
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	
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, you want to sort search results by last modified date.	String	<ul> <li>Title - The title of the content block</li> <li>ID - The XML content block ID number</li> <li>Date Created - The date the content block was created</li> <li>Date Modified - The date the content block was last modified</li> <li>AuthorLName - The last author's last name</li> <li>AuthorFname - The last author's first name</li> <li>StartDate - 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	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. To access the window click the gray button.		For more information on SearchParmXML, see "SearchParmXML Property" on page 102
ShowSearchBoxAlways	If set to <b>false</b> , the search box does not appear on PostBack	Boolean	True False

Property	Description	Data Type	Values
Weighted	If you set this value to <b>true</b> , content blocks are ranked by the number of criteria matches. 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. 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.	Boolean	<ul> <li>True - display every content block that matches <i>at least one</i> search criteria</li> <li>False - show only content blocks that match <i>all</i> search criteria</li> </ul>
WrapTag	Allows a developer to specify a server control's tag. The default is <b>Span</b> .	String	<ul> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>
xmlConfigID	ID of the XML configuration 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 32

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	string
Search Parameter /root/pdate Select One	date
Search Parameter	decimal
Search Parameter <pre>/root/authors/author/lastname ///initial Select One //initial Sele</pre>	string

3. Click the check boxes for the parameters you want to search with.

string
data
uale
decimal
string

4. Define the parameter.

🖶 Build Search Parameter	
Search Parameter	string
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.

- 5. Click OK.
- 6. Build the Web form.
- 7. 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

Νοτε

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.
- Drag and drop an IndexSearch Server Control on your Web form.
- 3. Set the properties in the properties window.

Νοτε

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.

[Literal "Literal1"] IndexSearchBlock[XmlConfigId=4]		
roperties		
IndexSearch1 Ektron.Cms.	Controls.IndexSearch	
Ektron		
Authenticated	True	
ButtonText	Search	
ContentParameter		
DisplayXslt	ecmNavigation	
EmptyResultMsg	Your search did not match any	
FolderId	0	
Hide	True	
HitsMarker	*	
Labelafter	After	
Labelbefore	Before	
LabelBetween	Between	
LabelContains	Contains	
LabeldateBetween	Between	
LabelEqual	Equal	
LabelExactPhrase	Exact Phrase	
LabelGreaterThan	Greater Than	
Labelinputdate	Input needs to be a date	
Labelinputnumber	Input needs to be a number	
LabelLessThan	Less Than	
Labelon	On	
LabelSelect	No Selection	
Language	1033	
OrderBy	Title	
OrderByDirection	Descending	
Recursive	True	
SearchParmXML		
ShowSearchBoxAlways	False	
Weighted	True	
XmlConfinId	4	

#### 5. Add the following code to the code behind.

```
IndexSearchl.Parm.XPath = "/root/subject"
Dim arStr As String()
Dim Str As String
Dim strRet As String
arStr = IndexSearchl.GetXPathValues("/root/subject")
For Each Str In arStr
strRet += Str & "<br/>>"
Next
Literall.Text = strRet
```

Here is a description of what is happening in the code.

- This line defines the parameter XPath location: IndexSearch1.Parm.XPath = "/root/subject"
- The next three lines create variables:

```
Dim arStr As String()
Dim Str As String
Dim strRet As String
```

- This line set the variable arStr as an array of the XPathValues: arStr = IndexSearch1.GetXPathValues("/root/ subject")
- The for next loop iterates through the array of XPathValues:

```
For Each Str In arStr
    strRet += Str & "<br/>>"
Next
```

- The last line displays the values on your web form: Literal1.Text = strRet
- 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.EkSearch.XMLDataType.String
      IndexSearch1.Parm.SearchType =
Ektron.Cms.Common.EkSearch.XMLSearchRangeType.ExactPhrase
      IndexSearch1.Parm.Value1 = Request.QueryString("publisher")
      bRet = IndexSearch1.AddParm()
      IndexSearch1.Search()
      Literall.Text = "<hl>Publisher " & Request.QueryString("publisher") &
- Books < /h1>
      Literal2.Text = "<a href=""webform.aspx"">Back</a>"
   Else
      Literall.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=" &</pre>
Server.UrlEncode(Str) & """>" & Str & "</a><br/>>"
      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.

🕴 Address 💩 http://localhost/CMS400Example/webform.aspx?publisher=Prentice+Hall 🛛 💉 🍋 G	60
Einks 💰 EktronNet Welcome 💰 Ektron, Inc. 💰 CMS400 Example	»
<b>Publisher Prentice Hall - Books</b> <u>Visual Basic NET How to Program</u> 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.	
Simply Visual Basic.NET 2003 An Application-Driven Tutorial Approach	
Back	>
🝘 Done 🧐 Local intranet	:

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()

## ListSummary Server Control

The List Summary Server Control is used to display a list of content blocks on a Web page.

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. The List Summary Server Control properties are described in this table.

This subsection contains the following topics:

- "ListSummary Server Control Properties" on page 114
- "List Summary Example" on page 118
- "OrderKey Property" 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.Net 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>FolderPath</b> property. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	String	
ContentParameter	Checks the QueryString for this value and replaces the collection with a content block when specified. Leave blank to always display the Collection or List Summary.	String	
Direction	How to order the hyperlinks on the list. The sort field is determined by the OrderKey property.	either ascending or descending	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.

Property	Description	Data Type	Values
DisplayXslt	Determines how the information on the page is displayed	String	None-databind only ecmNavigation - lists the title of every content block in the folder See Also: "Example of ecmNavigation Display" on page 121 ecmTeaser - lists the title of every content block in the folder plus the content summary See Also: "Example of ecmTeaser Display" on page 122 Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page
FolderPath	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 32 The Recursive property determines whether content blocks in this folder's child folders also appear.	String	Any folder
Hide	Used to hide output of List Summary in design time and run time.	Boolean	True = Hide results False = show results
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	Integer
MaxResults	Maximum number of summaries displayed on a Web page.	Integer	any integer 0 = unlimited

Property	Description	Data Type	Values
OrderKey	Sort the list by one of the values.	must be one of the values	Title DateModified DateCreated LastEditorFname LastEditorLname See Also: "OrderKey Property" on page 117
Random	Set to <b>True</b> if you want to randomly display one content block in the specified folder. The content changes each time a user views the page.	Boolean	True False
Recursive	Determines if the display includes content blocks in child folders of the selected folder.	Boolean	True False
WrapTag	Allows a developer to specify a server control's tag. The default is <b>Span</b> .	String	<ul> <li>Span - The <span> tag is used to designate an inline portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>

#### **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.	, RC Sportster. The RC Sportster is design mid level RC racer currently on the market nference (8/27/2002 2:44:44 PM) growing company in the PC host the 3rd annual RC conference in Atla
DateCreated	Displays the date when the content was first created.	d level RC racer currently on the mark erance (4/25/2002 4:42:55 PM) owing company in the RC ost the 3rd annual RC conference in A
LastEditorFname	Displays the first name of the user who last edited the content block.	nid level RC racer currently nference <i>(John)</i> growing company in the Ru host the 3rd annual RC co
LastEditorLnames	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 <i>(Doe)</i> st growing company in the R to host the 3rd annual RC co

#### List Summary Example

Below is an example of the List Summary Server Control properties.

Ξ	Ektron			
	Authenticated	True		
	ContentParameter			
	Direction	Ascending		
	DisplayXslt	ecmNavigation		
	FolderPath	\Marketing\News		
	Hide	False		
	Language	1033		
	MaxResults	0		
	OrderKey	Title		
	Random	False		
	Recursive	False		

The example has the following properties:

- Direction The results are displayed in ascending order by title
- FolderPath Displays summaries from the \Marketing\News folder
- MaxResults Unlimited results appear
- OrderKey Summaries sorted by content block title
- Random Displays summary for all content blocks in folder, not just one random one
- Recursive Includes only selected folder, not its child folders
- Xslt will use the navigation display format
# **Collection Server Control**

The Collection Server Control is used to display a list of content block links that were created as a collection in the Ektron CMS400.NET Workarea. The control is highly customizable, allowing you to define 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 properties.

Human Resources Forms

While you were out...

Employee Forms

Absence Request Form

Direct Deposit Form

Manager Forms

Personnel Action Form

This subsection contains the following topics:

- "Collection Server Control Properties" on page 120
- "Example of ecmNavigation Display" on page 121
- "Example of ecmTeaser Display" on page 122
- "Using the Collection Server Control Programmatically Example" on page 123

## **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.Net 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 32	String
ContentParameter	Checks the QueryString for this value and replaces the collection with a content block when specified. Leave blank to always display the Collection or List Summary.	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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer
DisplayXslt	Determines how the information on the page is displayed None-databind only ecmNavigation - lists the title of every content block in the collection See Also: "Example of ecmNavigation Display" on page 121 ecmTeaser - lists the title of every content block in the collection plus the content summary See Also: "Example of ecmTeaser Display" on page 122 Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page	String

Property	Description	Data Type
DynamicParameter	To make this collection dynamic, select <b>coll_id</b> . When you do, this server control uses the collection passed as a URL parameter.	String
GetHtml	Set to <b>True</b> 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 DataGrid.	Boolean
Hide	Used to hide output of collection in design time and run time. <b>True</b> = Hide collection <b>False</b> = Display collection	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
Random	Set to <b>True</b> if you want to randomly display one collection item. The item changes each time a user views the page.	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 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.</div></span>	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="/">

</a>
```

### 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
------	----------	---------	-------------	---------	--------	----------

#### **Employment Opportunities**

Plastic Molder #123

RC International is looking for an experienced plastics molder.

#### Servo Control Engineer #124

RC International is looked for a highly skilled servo control engineer to join our team.

#### 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="/">
</tsple>
```

</xsl:for-each></xsl:template></xsl:stylesheet>

## 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**.

Start Page randomcontentldisplay.aspx* randomcontent_htmldisplay.aspx.vb*						
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	Properties					
	Collection1 Ektron.Cr	ms.Controls.Collection 🔹				
	1 2 I II II					
	Ektron	×				
	Authenticated	True				
	CmsSitePath	http://localhost/CM5400e				
	ContentParameter					
	DefaultCollectionⅢ	5				
	DisplayXslt					
	DynamicParameter					
	GetHtml	True				
	Hide	False				
	Language	1033				
	Random	True				
	🗆 Layout					
	Height					
	Width					
	🛛 Misc					
	(ID)	Collection1 👻				
	Properties ? Dyr	namic Help				

3. Add the following code to the code behind.

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

4. Build and browse your web form.



# **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 130

You can adjust a calendar's look and feel by using a style sheet. See Also: "Calendar Style Sheet" on page 131 and "calendarStyles.css Class Description" on page 132.

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	Calenc	lar display					
Logged in							
	vember		De	cember 20	004		January
	Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1 🖸 🔍	2 88 99	3 #4 94	4 80 94
		6 <b>• a Q</b>	7 🖬 🔍	8 🖪 🔍	9 🛃 🔍	10 .	11 🖪 🔍
	<u>a</u> Q.	13 🖪 🔍	14 🖪 🔍	15 🖪 🔍	16 🔽	17 🖪 🔍	18 🖪 🔍
		20 🖪 🔍	21 6	22 🖪 🔍	23 🖪 🔍	24 <b>• 6</b>	25 🖪 🔍
	a Q	27 • •	28 🖪 🔍	29 🖪 🔍	30 🖪 🔍	31	

Login Status	Calenda	ar display					
Logged Out							
	ovember		D	ecember 20	004		Janua
	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 128
- "Calendar Style Sheet" on page 131
- "calendarStyles.css Class Description" on page 132

## **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.Net 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 32	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. <i>See Also:</i> "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer

Property	Value	Data Type
Property DisplayType	Value         Specify the Calendar view type. The default is Month.         • Month - A 30 day calendar month view.         Image: A straight of the test of test of the test of	Data Type String
	<ul> <li>Inline - Display events that occur from the start date of the calendar through the end date.</li> <li>Sunday, May 01, 2005 - Saturday, July 30, 2005</li> <li>Sunday, May 01, 2005 - Saturday, July 30, 2005</li> <li>Sunday, May 01, 2005 - Saturday, July 30, 2005</li> <li>Work 0200700</li> <li>Work</li></ul>	
DynamicParameter	To make this calendar dynamic, select <b>calendar_id</b> .	String
	When you do, this server control uses the calendar passed as a URL parameter.	

Property	Value	Data Type
EndDate	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.	Date
Hide	Used to hide output of calendar in design time and run time. <b>True</b> = Hide calendar <b>False</b> = 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 <b>False.</b> <b>True</b> = Show raw XML data <b>False</b> = 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
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an inline portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	String

## Calendar Style Sheet

Before you insert a Calendar into your Ektron CMS400.NET template, Ektron recommends referencing the style sheet \webroot\cms400example\calendarStyles.css in your template. Copy the style sheet to your webroot folder then add it to your project. Next create a reference to calenderStyles.css. For example:

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.

### calendarStyles.css Class Description

The following table describes the classes found in the calendarStyles.css file. This table contains the following sections.

- "Single Day View (Day)" on page 132
- "Month View (month)" on page 136
- "Inline View (inline)" on page 148
- "Month-Up with Event (monthupwithevent)" on page 150
- "Event Type Control (used in month display)" on page 157

Style Class	Description	Attributes and Default Settings
Single Day View		

Style Class	Description	Attributes and Default Settings
.dv_HourCell	The workday hour cells. The following example shows these cells with a red background.           << 5/4/2005	<ul> <li>background: #99DDFF</li> <li>font-family: Arial, Arial Narrow, Times New Roman</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> </ul>
.dv_HourCellEve	The evening and morning hour cells. The following example shows these cells with a red background.         << 5/4/2005	<ul> <li>background: #6699DD</li> <li>font-family: Arial, Arial Narrow, Times New Roman</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> </ul>

Style Class	Description	Attributes and Default Settings
.dv_EventTitle	An event listed on the daily calendar. The following example shows the event with a red background.	<ul> <li>background: #aaddaa</li> <li>font-family:</li> </ul>
	Inursday, May 05, 2005         Inursday, May 05, 2005         Inursday, May 05, 2005           12:00 AM         1:00 AM         1:00 AM           2:00 AM         3:00 AM         1:00 AM           5:00 AM         5:00 AM         1:00 AM           5:00 AM         1:00 AM         1:00 AM           5:00 AM         1:00 AM         1:00 AM           10:00 AM         1:00 AM         1:00 AM           11:00 AM         1:00 PM         1:00 PM           2:00 PM         1:00 PM         1:00 PM           1:00 PM         1:00 PM         1:00 PM	Arial, Arial Narrow, Times New Roman • font-size: 9pt • font-weight: bold • cursor: pointer
.dv_BorderCell	The borders of the hour cells. The following example shows the borders in red.	• background: #0033AA
	<< 5/4/2005         Thursday, May 05, 2005         5/6/2005 >           12:00 AM	

Style Class	Description	Attributes and Default Settings
.dv_DayHeaderNext	The button for moving the calendar to the next day. The following example shows the day header next in red.         <<<5/4/2005	<ul> <li>background: #99DDFF</li> <li>color: #0033AA</li> <li>font-family: Arial</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> <li>text-align: right</li> </ul>
.dv_DayHeaderPrev	The button for moving the calendar to the previous day. The following example shows the day header previous in red.         Thursday, May 05, 2005       \$16/2005.>>         12:00 AM       \$100 AM         2:00 AM       \$100 AM         3:00 AM       \$100 AM         5:00 AM       \$100 AM         9:00 AM       \$100 AM         10:00 AM       \$100 AM         5:00 AM       \$100 AM         10:00 AM       \$100 AM         10:00 AM       \$100 AM         10:00 AM       \$100 PM         2:00 PM       \$100 PM         3:00 PM       \$100 PM         5:00 PM       <	<ul> <li>background: #99DDFF</li> <li>color: #0033AA</li> <li>font-family: Arial</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> <li>text-align: right</li> </ul>

Style Class	Description	Attributes and Default Settings
.dv_DayHeader	The day header for the calendar. The following example shows the day header in red.	<ul> <li>background: #99DDFF</li> <li>color: #0033AA</li> <li>font-family: Arial</li> <li>font-size: 11pt</li> <li>font-weight: bold</li> <li>text-align: center</li> </ul>
Month View (mo	onth)	

Style Class	Description	Attributes and Default Settings
.mv_weekendDOWHeader	Weekend day of week header. The following example shows the weekend day of week header in red.	<ul> <li>Default Settings</li> <li>background: #c5daef</li> <li>font-family: arial</li> <li>font-weight: bold</li> <li>font-size: 8pt</li> <li>border-top: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> <li>border-right: #c9bda5 1px</li> </ul>
		solid <b>border-left</b> : #c9bda5 1px solid <b>text-align</b> : center

Style Class	Description	Attributes and Default Settings
.mv_weekdayDOWHeader	Weekday day of week header. The following example shows the weekday day of week header in red.	background:     #c5daef     fort family:
	ss.April May June 22	• font-family:
	Sunday         Manday         Taxaday         Wednesday         Thanaday         Ender         Saturday           50         0         1         50         2         50         2         50         4         50         5         50         5         50         2         5         50         2         5         50         2         50         2         1	<ul> <li>font-weight:</li> <li>bold</li> </ul>
	Image: Non-State         Image: Non-State<	• font-size: 8pt
	Cold Cold Cold         Sector         Sector         Formation           Cold Cold Cold         Sold Cold         So	• <b>border-top</b> : #c9bda5 1px solid
	6 9 22 6 9 23 76 9 24 76 9 25 76 9 26 76 9 27 76 9 28	• <b>border-bottom</b> : #c9bda5 1px solid
		• <b>border-right</b> : #c9bda5 1px solid
		<ul> <li>border-left: #c9bda5 1px solid</li> <li>text-align: center</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_MonthHeader	Month header. The following example shows the month header background in red.	<ul> <li>font-weight: bold</li> <li>font-size: 12pt</li> <li>background: transparent</li> <li>font-family: Arial</li> </ul>
.mv_MonthHeaderBkg	Month header background. The following example shows the month header background in red.	<ul> <li>border-right: #c9bda5 2px solid</li> <li>border-top: #c9bda5 2px solid</li> <li>border-left: #c9bda5 2px solid</li> <li>font-reight: bold</li> <li>font-size: 9pt</li> <li>background: #A5B7C9</li> <li>font-family: arial</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_MonthHeaderNext	The button for moving the calendar to the next month. The following example shows the month header next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to the next background in red.         Image: Comparison of the calendar to	<ul> <li>font-weight: bold</li> <li>fontsize: 9pt</li> <li>background: transparent</li> <li>font-family: Arial</li> <li>text-align: right</li> </ul>
.mv_MonthHeaderPrev	The button for moving the calendar to the previous month. The following example shows the month header previous background in red.	<ul> <li>font-weight: bold</li> <li>fontsize: 9pt</li> <li>background: transparent</li> <li>font-family: Arial</li> <li>text-align: left</li> </ul>

Style Class	Description	Attributes and Default Settings	
.mv_NonMonthBorder	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.	<ul> <li>border-right: #c9bda5 1px solid</li> <li>border-top: #c9bda5 1px solid</li> <li>background:</li> </ul>	
	• • • • • • • • • • • • • • • • • • •	<ul> <li>background. #e0e0e0</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> </ul>	
.mv_DateCell	The cells where the numerical weekdays appear.	• font-weight:	
	The following example shows the date cell in red.         Image: Image <th comest:="" im<="" image:="" td=""><td><ul> <li>bold</li> <li>font-size: 9pt</li> <li>background: #F0E9C5</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> <li>font-family: arial</li> </ul></td></th>	<td><ul> <li>bold</li> <li>font-size: 9pt</li> <li>background: #F0E9C5</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> <li>font-family: arial</li> </ul></td>	<ul> <li>bold</li> <li>font-size: 9pt</li> <li>background: #F0E9C5</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> <li>font-family: arial</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_DateCellWeekend	The cells where the numerical weekend days appear. The following example shows the date cell weekend in red.	<ul> <li>font-weight: bold</li> <li>font circu</li> </ul>
	< <a href="https://www.autor.org/autor.org/linearing"></a>	<ul> <li>font-size:</li> <li>9pt</li> </ul>
	Image: Solution of the second seco	• background: #C5DAEF
	Te S S 2 56 S 10 Te S 11 Te S 12 Te S 13 Te S	<ul> <li>border-left: #c9bda5 1px solid</li> </ul>
	26 Q C 6 Q 16 26 Q 17 26 Q 18 26 Q 19 26 Q 28 26 Q	<ul> <li>border-bottom: #c9bda5 1px solid</li> </ul>
	ଅଳ ସା <mark>ଅ</mark> ଅଳ ସା 22 ଅଳ ସା 24 ଅଳ ସା 25 ଅଳ ସା 28 ଅଳ ସା 27 ଅଳ ସା <mark>ଅ</mark>	<ul> <li>font-family: arial</li> </ul>
.mv_DateCellToday	The cells where today's numerical day appears. The following example shows the date cell today in red.	• background: #FFE354
	ss Illarch April Illar >>	<ul> <li>border-left: #b38b3b 1px</li> </ul>
	Sunday Monday Tuesday Wednesday Thursday Friday Saturday	solid
		<ul> <li>border-bottom: #b38b3b 1px</li> <li>solid</li> </ul>
		<ul> <li>font-family: Arial</li> </ul>
	56 S. 10 56 S. 11 56 S. 12 56 S. 14 56 S. 15 56 S. 16	• font-size: 9pt
	56 3, 17 56 3, 18 56 3, 19 56 3, 20 56 3, 21 56 3, 22 56 3, 23	<ul> <li>font-weight: bold</li> </ul>
	Ce C 24 Ce C 25 Ce C 20 Ce C 27 Ce C 20 Ce C 29 Ce C 20	

Style Class	Description	Attributes and Default Settings
.mv_DateCellEmptyWeekend Note: These cells appear empty when you are not logged in to Ektron CMS400.NET.	The weekend cells that hold the Add Event and view Event buttons. The following example shows the date cell empty weekend in red.	<ul> <li>background: #C5DAEF</li> <li>border-bottom: #c9bda5 1px solid</li> <li>font-family: Arial</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> </ul>
.mv_DateCellEmptyToday Note: This cell appears empty when you are not logged in to Ektron CMS400.NET.	The today cell that holds the Add Event and View by the today in red.	<ul> <li>background: #FFE354</li> <li>border-bottom: #B38B3B 1px solid</li> <li>font-family: Arial</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_DateCellEmpty Note: These cells appear empty when you are not logged in to Ektron CMS400.NET	The weekday cells that holds the Add Event and yiew Event buttons. The following example shows the date cell empty in red.	<ul> <li>background: #f0e9c5</li> <li>border-bottom: #c9bda5 1px solid</li> <li>font-family: Arial</li> <li>font-size: 9pt</li> <li>font-weight: bold</li> </ul>
.mv_eventContainer	The popup window that appears when the event is hovered over. The following example shows the event container circled in red.	<ul> <li>visibility: hidden</li> <li>position: absolute</li> <li>Note: When the visibility property is set to visible, the event container appears upon page load.</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_dayBorderToday	The today cell that holds listed events. The following example shows the day border today in red.	• <b>border-right</b> : #b38b3b 2px solid
	Sindey         Monday         Tuenday         Mednessing         Futures         <	<ul> <li>border-top: #b38b3b 2px solid</li> <li>background: #ffe354</li> <li>border-left: #b38b3b 2px solid</li> <li>border-bottom: #b38b3b 2px solid</li> </ul>
.mv_dayBorder This class effects all weekday cells, except the today cell.	The weekday cell that holds listed events. The following example shows the day border in red.	<ul> <li>border-right: #c9bda5 1px solid</li> <li>border-top: #c9bda5 1px solid</li> <li>background: #f0e9c5</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_dayBorderWeekend	The Weekend day cell that holds listed events. The following example shows the day border weekend in red.	• <b>border-right</b> : #c9bda5 1px solid
	Security         Monday         Total and	<ul> <li>border-top: #c9bda5 1px solid</li> <li>background: c5daef</li> <li>border-left: #c9bda5 1px solid</li> <li>border-bottom: #c9bda5 1px solid</li> </ul>
.mv_eventTitle	The event title cell. The following example shows the event title cell in red.	• font-weight: normal
	<< April May June >>	• font-size
	Sunday Monday Turesday Wednesday Thureday Friday Saturday	<ul> <li>9pt</li> <li>background: transporent</li> </ul>
	6 9 8 6 9 6 9 10 6 9 11 6 9 12 6 9 13 6 9 14	• font-family: Arial
	6 Q 15 76 Q 18 76 Q 17 76 Q 18 76 Q 19 76 Q 20 76 Q 21	
	Té Q 22 Té Q 22 Té Q 24 Té Q 25 Té Q 28 Té Q 27 Té Q 28	

Style Class	Description	Attributes and Default Settings
.mv_DayCellToday	Today's event title cell. The following example shows the day cell today in red.	<ul> <li>background: transparent</li> <li>font-family: Arial</li> <li>font-weight: bold</li> <li>font-size: 9pt</li> </ul>
.mv_TodayCell	The font for the event pop-up. The following example shows the today cell font in red.	<ul> <li>font-weight: normal</li> <li>font-size: 9pt</li> <li>background: #C5DAEF</li> <li>font-family: Arial</li> </ul>

Style Class	Description	Attributes and Default Settings
.mv_eventPopup	The border for the event pop-up. The following example shows the event pop-up border in red.	<ul> <li>border-right: #000000 1px solid</li> <li>border-top: #000000 1px solid</li> <li>border-left #000000 1px solid</li> <li>border-bottom: #000000 1px solid</li> </ul>
Inline View (inlin	ne)	1
.ILViewHeaderBkg	The inline view header background. The following example shows the view header background in red. Friday, April 01, 2005 - Wednesday, June 01, 2005 4/12/2005 Football Shafer Stadium Start 12:00 AM End: 12:00 AM Start 12:00 AM End: 12:00 AM End: 12:00 AM Manchester Monarchs take on the Portland Pirates	<ul> <li>border-right: #9999ff 2px solid</li> <li>border-top: #9999ff 2px solid</li> <li>font-weight: bold</li> <li>font-size: 11pt</li> <li>background: #ddddff</li> <li>border-left: #9999ff 2px solid</li> <li>font-family: Arial</li> <li>text-align: center</li> </ul>

Style Class	Description	Attributes and Default Settings
.ILViewEvent	Font for the event description. The following example shows the view event font in red. Friday, April 01, 2005 - Wednesday, June 01, 2005 4/12/2005 Football Shafer Stadium Start 12:00 AM End: 12:00 AM Start 12:00 AM End: 12:00 AM End: 12:00 AM	<ul> <li>font-size: 9pt</li> <li>font-family: Arial</li> </ul>
.ILViewDate	The date for the inline view. The following example shows the view date background in red. Friday, April 01, 2005 - Wednesday, June 01, 2005 4/12/2005 Football Shafer Stadium Start 12:00 AM End:12:00 AM Start 12:00 AM End:12:00 AM End:12:00 AM End:12:00 AM	<ul> <li>border-right: #9999ff 1px solid</li> <li>border-top: #9999ff 1px solid</li> <li>font-weight: bold</li> <li>font-size: 9pt</li> <li>background: #ddddff</li> <li>border-left: #9999ff 1px solid</li> <li>font-family: Arial</li> </ul>

Style Class	De	scrij	otio	n	Attributes and Default Settings					
.ILViewDayCell	The exa Fr 4/12/2 Footb Shate Start End 1 5/5/20 Hock VWA Start End 1 Manc	e cell ample iday, 2005 all r Stad 12:00 A 12:00 A 12:00 A hester	the e sho April M M M M M M M Monar	event:	<ul> <li>border-right: #9999ff 1px solid</li> <li>border-top: #9999ff 1px solid</li> <li>font-weight: bold</li> <li>font-size: 9pt</li> <li>background: #eeeeff</li> <li>border-left: #9999ff 1px solid</li> <li>border-bottom: #9999ff 1px solid</li> <li>font-family: Arial</li> </ul>	t t b				
Month-Up with	Ever	nt (r	nor	nthu	pw	ith	eve	ent)		
.mup_nonMonthCell	The do sho Sun 1	e cells not co ows th Apr Mon 2	s at thontaine no	he be n day on-mc y, 200 Wed 4	<ul> <li>background: #eeeeee</li> <li>font-family: Arial</li> <li>font-weight: normal</li> <li>font-size:</li> </ul>	al				
	15       16       17       18       19       20       21       8pt         22       23       24       25       26       27       28       Manchester       center         29       30       31       10       10       10       border-rigg         2004        2005        2006       border-rigg									
									<ul> <li>border-bottom: #aaaaaa 1px solid</li> <li>border-left: #aaaaaa 1px solid</li> </ul>	

Style Class	Description	Attributes and Default Settings
.mup_weekendCell	The weekend day cell. The following example shows the weekend cell in red. Sun Mon Tue Wed Thu Fri Sat       5/5/2005         Hockey Game       Hockey Game         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       Manchester         29       30       31       1       1       1       10       10       11         2004 - 2005 - 2006       2005       10       10       10       10       10       10	<ul> <li>background: #dedeff</li> <li>font-family: Arial</li> <li>font-weight: normal</li> <li>font-size: 8pt</li> <li>text-align: center</li> <li>border-right: #aaaaaa 1px solid</li> <li>border-bottom: #aaaaaa 1px solid</li> <li>border-left: #aaaaaa 1px solid</li> </ul>
.mup_weekdayCell	The week day cell. The following example shows the week day cell in red.         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       Manchester         29       30       31       0       0       0       0       0       0         2004 - 2005 - 2006       2005       2006       0       0       0       0       0	<ul> <li>background: #eeffff</li> <li>font-family: Arial</li> <li>font-weight: normal</li> <li>font-size: 8pt</li> <li>text-align: center</li> <li>border-right: #aaaaaa 1px solid</li> <li>border-bottom: #aaaaaa 1px solid</li> <li>border-left: #aaaaaa 1px solid</li> </ul>

Style Class	De	scrip	otior	ו	Attributes and Default Settings						
.mup_weekdayEventCell	Hig follo in re	hligh owing ed.	ts the J exa	e wee mple	• background: #ffffee						
	<<	Apr	May	y, 200							
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	• font-weight:		
	1	2	3	4	- 6	6	7	Location. V VVA	normai		
	8	9	10	11	12	13	14		<ul> <li>font-size: 8pt</li> </ul>		
	15	16	17	18	19	20	21	Manchastar	• text-align: center		
	22	23	24	25	26	27	28	Monarchs take on	• border-right:		
	29	30	31		-			the Portland Pirates	#aaaaaa 1px		
	11155	20	04 2	2005	2006	2			solid		
									• border-bottom: #aaaaaa 1px solid		
									<ul> <li>border-left: #aaaaaa 1px solid</li> </ul>		
									• cursor: pointer		
.mup_weekendDOWHeader	We exa in r	eken Imple ed.	d day shov	y of w ws the	veek e we	hea eke	der. nd c	The following lay of week header	<ul> <li>background:</li> <li>#ddddff</li> </ul>		
	<apr 2<="" may="" th=""><th colspan="3">v. 2005</th><th>5/5/2005</th><th>• font-family: Arial</th></apr>				v. 2005			5/5/2005	• font-family: Arial		
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	• font-weight:		
	1	2	3	4	5	6	7	Location:VWA	Bold		
	8	9	10	11	12	13	14		• font-size: 8pt		
	15	16	17	18	19	20	21		border-right;		
	22	23	24	25	26	27	28	Manchester Monarcha taka on	#aaaaaa 1px		
	29	30	31					the Portland Pirates	solid		
		20	<u>04</u> 2	2005	200	2			• border-bottom: #aaaaaa 1px solid		
									border-left:     #aaaaaa 1px     solid		
									• text-align: center		

Style Class	De	scrip	otior	ì	Attributes and Default Settings				
.mup_weekdayDOWHeader	We exa in r	ekda <u>y</u> Imple ed.	y day shov	∕ of w ws the	• background: #eeeeff				
	<<	Apr	May	y, 200	5	Jun	>>	5/5/2005	• font-family: Ariai
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	• font-weight:
	1	2	3	4	5	6	7	Location: VVVA	BOID
	8	9	10	11	12	13	14		<ul> <li>font-size: 8pt</li> </ul>
	15	16	17	18	19	20	21	M	border-right:
	22	23	24	25	26	27	28	Monarchs take on	#aaaaaa 1px
	29	30	31					the Portland Pirates	SOIIC
						<ul> <li>border-bottom. #aaaaaa 1px solid</li> <li>border-left: #aaaaaa 1px solid</li> <li>text-align: center</li> </ul>			
.mup_yearFooter	The sho	e yeai ws th	sele le ye	ection ar foc	cell. ter i	The n re	e foll d.	lowing example	• background: #eeeeee
	<<	Apr	Ma	y, 200	5	Jun	>>	5/5/2005	• font-family:
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Location:VWA	Arial
	1	2	3	4	5	6	7		font-weight:
	8	9	10	11	12	13	14		Bold
	15	16	1/	18	19	20	21	Manchester	• font-size:
	22	30	31	20	20	21	20	Monarchs take on	9pt
		20	042	2005	2000			the Portland Pirates	• text-align:
									center

Style Class	De	scrip	otior	ו	Attributes and Default Settings				
.mup_monthHeaderBorder	The folle bor	e boro owing der ir	der a g exa n red.	round mple	• background: #eeeeee				
	<<	Apr	Ma	y, 200	5	Jun	>>	5/5/2005	Δrial
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	Allai
	1	2	3	4	5	6	7	Location:VWA	• font-weight:
	8	9	10	11	12	13	14		Bold
	15	16	17	18	19	20	21	1	font-size:
	22	23	24	25	26	27	28	Manchester	10pt
	29	30	31					Monarchs take on the Portland Pirates	• border-right:
		20	04 2	2005	2006	1			#aaaaaa 1px
									<ul> <li>border-top: #aaaaaa 1px solid</li> <li>border-bottom: #aaaaaa 1px solid</li> <li>border-left: #aaaaaa 1px solid</li> <li>border-left: center</li> </ul>
.mup_monthHeader	The the	e mor mon	nth he th he	eader ader	. The in rea	e foll d.	lowi	ng example shows	• background: #eeeeee
	<<	Apr	Ma	y, 200	)5	Jun	>>	5/5/2005	• font-family:
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	Arial
	1	2	3	4	5	6	7		• font-weight:
	8	9	10	11	12	13	14		Bold
	15	16	17	18	19	20	21	Manchaster	• font-size:
	22	23	24	25	26	27	28	Monarchs take on	10pt
	29	30	31					the Portland Pirates	• text-alion:
		20	<u>04</u> 2	2005	center				
Style Class	Description								Attributes and Default Settings
-----------------------------	--	--------------------------	--------------------------	-------------------------	--------------------------------	--------------------------	-----------------	--------------------------------	---
,mup_monthHeaderPrev	The folle pre	e prev owing vious	vious   exa   bacl	butto mple k grou	nth header. The onth header	• background: #eeeeee			
		âr:	Ma	y, 200	5/5/2005	Arial			
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	• fant weight
	1	2	3	4	5	6	7	Location. VVVA	<ul> <li>Bold</li> </ul>
	8	9	10	11	12	13	14		
	15	16	17	18	19	20	21		<ul> <li>font-size:</li> <li>8pt</li> </ul>
	22	23	24	25	26	27	28	Manchester Monarchs take on	
	29	30	31					the Portland Pirates	• text-align:
		20	04 2	2005.	. 2006	5			Center
.mup_monthHeaderNext	The next button, on the month header. The following example shows the month header next background in red.								<ul> <li>background: #eeeeee</li> <li>font-family:</li> </ul>
	~	Apr	Ma	iy, 20	05	1.0		Hockey Game	Arial
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Location:VWA	• font-weight:
	1	2	3	4	5	0	4	Bold	Bold
	0	16	17	12	10	20	21	-	• font-size:
	22	23	24	25	26	20	21	Manchester	8pt
	20	30	31	20	20	21	20	Monarchs take on	• text-align:
	23	20	0.4	2005	the Portland Pirates	center			
		21	104	2005					
.mup_eventDispBkg	The background for the event display. The following example shows the event display background in red.								background:     #eeeeff
deals with the background	<<	Apr	May	y, 200	)5	Jun	<u>&gt;&gt;</u>		<ul> <li>border-right.</li> <li>#aaaaaa 1px</li> </ul>
layer of the event display.	Sun	Mon	Tue	Wed	Thu	Fri	Sat		solid
the layer the event is	1	2	3	4	5	6	7		• border-ton <sup>.</sup>
displayed on.	8	9	10	11	12	13	14		#aaaaaa 1px
	15	16	17	18	19	20	21		solid
	22	23	24	25	26	27	28		• border-bottom:
	29	30	31						#aaaaaa 1px
		20	04 2	2005	2006				solid
									<ul> <li>border-left: #aaaaaa 1px solid</li> </ul>

Style Class	Description							Attributes and Default Settings			
.mup_eventDisplay	The event display. The following example shows the event display in red.							g example shows	background: transparent		
Note: .mup_eventDispBkg	<<.	Apr	Ma	y, 200	5/5/2005						
laver of the event display.	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game			
.mup_eventDisplay deals with	1	2	3	4	5	6	7	Location. V VVA			
the layer the event is	8	9	10	11	12	13	14				
displayed on.	15	16	17	18	19	20	21				
	22	23	24	25	26	27	28	Manchester Monarche take on			
	29	30	31					the Portland Pirates			
		20	04 2	2005	200	į					
.mup_eventDisplayDate	The exa	ever Imple	nt dis shov	play ws the	date e eve	hea ent c	der. lispl	The following ay date in red.	• font-size: 12pt		
	<			Jun	>>	5/5/2005	• tont-weight: bold				
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	background:		
	1	2	3	4	5	6	7	Location:VWA transparent • font-family: A • text-align: ce Manchester Monarchs take on the Portland Pirates • border-botton #aaaaaa 1px	transparent		
	8	9	10	11	12	13	14		<ul> <li>font-family: Arial</li> </ul>		
	15	16	17	18	19	20	21		• text-align: center		
	22	23	24	25	26	27	28		border-bottom:		
	29	30	31						#aaaaaa 1px		
		20	04 2	2005	solid						
.mup_eventDisplayEventTitle	Display for the event title. The following example shows the event display event title in red.						following example title in red.	• font-weight: bold			
	<<	Apr	Ma	y, 200	)5	Jun	>>	5/5/2005	background:		
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	transparent		
	1	2	3	4	5	6	7	Location. VWA	• font-family:		
	8	9	10	11	12	13	14		Arial		
	15	16	17	18	19	20	21	Manahaatar	• font-size:		
	22	23	24	25	26	27	28	Monarchs take on	10pt		
	29	30	31					the Portland Pirates	• text-align		
		20	04 2	2005	left						

Style Class	Description								Attributes and Default Settings
.mup_eventDisplayLocation	Dis exa	play f imple	or th show	e eve ws the	• background: transparent				
	<<	Apr	Ma	y, 200	)5	Jun	>>	5/5/2005	• font-family:
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	Arial
	1	2	3	4	5	6	7	Location. V WA	font-weight:
	8	9	10	11	12	13	14		normal
	15	16	17	18	19	20	21		• font-size
	22	23	24	25	26	27	28	Manchester Monarchs take on	10pt
	29	30	31					the Portland Pirates	• toxt-align:
		20	04 2	2005	2006	5			left
.mup_eventDisplayTimes	Dis shc	play f ws th	or ev ie ev	vent ti ent di	mes. ispla	. Th y tin	e fol nes	llowing example in red.	• <b>background</b> : transparent
	<<	Apr	May	y, 200	5	Jun	>>	5/5/2005	• font-family
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	Arial
	1	2	3	4	5	6	7	Start: 12:00 AM	font-weight:
	8	9	10	11	12	13	14	End:12:00 AM	normal
	15	16	17	18	19	20	21		• font-size
	22	23	24	25	26	27	28	Manchester 10pt	10pt
	29	30	31					the Portland Pirates	• toxt align:
		20	04 2	2005	2006	į		left	left
.mup_eventDisplayLongDesc	Dis follo des	plays owing cripti	the o exa on in	event mple red.	's lor shov	ng d vs tł	esci ne e	ription. The vent display long	background: transparent
	<<	Apr	Ma	y, 200	)5	Jun	>>	5/5/2005	<ul> <li>font-family:</li> <li>Arial</li> </ul>
	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Hockey Game	
	1	2	3	4	5	6	7	Location: VVVA	• font-weight:
	8	9	10	11	12	13	14	1	nomai
	15	16	17	18	19	20	21		• font-size:
	22	23	24	25	26	27	28	Manchester	10pt
	29	30	31	1111				Monarchs take on the Portland Pirates	text-align:
		20	04 2	2005	2000	5		in the second second	left
Event Type Con	ntrol	(us	ed i	in m	ont	th c	dis	play)	

Style Class	Description	Attributes and Default Settings
.etCtrl_Background	The background of the event type control cell. The following example shows the event type control background in red.	<ul> <li>border-right: #c9bda5 2px solid</li> <li>border-top:</li> </ul>
	B 2 10 11 12 12 14	#c9bda5 2px solid • border-left:
	15 18 12 18 19 20 21 Rodd-Enid Holyw	#c9bda5 2px solid • border-bottom:
	Charoliseship 22 23 24 25 28 22 28	#c9bda5 2px solid ● font-size:
	22 20 23	9pt • background: #A5B7C9
	Ciri Alick to Show Freek Type: Hockay Fast ball Backet ball Field Hockey	<ul> <li>font-family: Arial</li> </ul>
.etCtrl_InstructCell	The instructions cell for event type control. The following example shows the instruction cell in red.	<ul> <li>background: transparent</li> <li>font formility</li> </ul>
	Wethwardsy         Manday         Tuescley         Wethwardsy         Theready         Fristry         Saturday           3         2         2         4         5         5         Z         Z           4         2         2         4         5         5         Z         Z	<ul> <li>font-family: Arial</li> <li>font-size: 9pt</li> </ul>
		• font-weight: bold
	Bodd Endd         Highar           1100ar         Charrolanshin           22         23         24         25         28         27         28	<ul> <li>text-align: right</li> <li>vertical-align:</li> </ul>
	22 22 22	top
	CBA-Click to Show Event Fysk! Hockey Plact ball Field Hockey	



# **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.

Products

 RC Cars
 <u>RC Cheetah</u>
 <u>RC Sportster</u>
 RC Planes
 <u>RC Lilly</u>
 <u>RC Redstar</u>
 <u>Products Page</u>
 <u>Visit Ektron.com</u>

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 vs. DhtmlMenu Server Control" on page 160
- "Menu Server Control Properties" on page 161
- "Using DisplayXslt Samples" on page 163

#### Menu Server Control vs. DhtmlMenu Server Control

The Menu Server Control calls a menu to be displayed on a page. You can manipulate the menu by using the DisplayXslt property. With the DhtmlMenu Server Control, you display a dynamic menu with drop down boxes and submenus. You manipulate DhtmlMenus with a style sheet. DhtmlMenus save space on the Web page by having the drop down menus and submenus float over content.

**WARNING!** You cannot create a DHTML menu using the code behind -- you must drag and drop it. Once the menu is on the template, you can manipulate its properties using the code behind.



#### 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.Net 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 32	String

Property	Value	Data Type
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. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer
DisplayXslt	The Xslt to use to render the menu. <b>None</b> - databind only SampleMenu - A sample display, formatted as a bulleted menu list TreeMenu - A sample display, formatted as a folder tree. You can expand the tree by clicking on the folder icon. <b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page For more information on using the SampleMenu and TreeMenu DisplayXslt, see "Using DisplayXslt Samples" on page 163.	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. <b>True</b> = Hide menu <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).	Integer
WrapTag	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an inline portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	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.



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 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".



#### SampleMenu XSL code

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

```
<LI>
```

```
<xsl:apply-templates select="Menu"/>
```

```
</LI>
```

```
</xsl:template>
```

<xsl:template name="makelist" match="Menu">

```
<xsl:choose>
```

```
<xsl:whentest="Link=''"><xsl:value-of select="Title"/></
xsl:when><xsl:otherwise>
```

```
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',Link)"/></xsl:attribute>
```

<xsl:value-of select="Title"/></A></xsl:otherwise></xsl:choose>

<UL><xsl:apply-templates select="Item"/></UL>

</xsl:template>

```
<xsl:template match="Item">
```

```
<xsl:param name="pItemLink"><xsl:value-of select="ItemLink"/></
xsl:param>
```

<xsl:choose><xsl:whentest="ItemLink=''">

<LI>

<xsl:value-of select="ItemTitle"/>

<xsl:apply-templates select="Item | Menu"/></LI>

```
</xsl:when>
```

<xsl:otherwise>

```
<LI><A>
```

```
<xsl:choose>
```

<xsl:whentest="ItemType='ExternalLink'">

```
<xsl:attribute name="href"><xsl:value-of select="ItemLink"/></
xsl:attribute>
```

<xsl:value-of select="ItemTitle"/>

</xsl:when>

```
<xsl:whentest="ItemType='Javascript'">
```

```
<xsl:attribute name="onClick">Javascript:<xsl:value-of</pre>
select="ItemLink"/></xsl:attribute>
             <xsl:attribute name="href">#</xsl:attribute>
            <xsl:value-of select="ItemTitle"/>
         </xsl:when>
         <xsl:otherwise>
            <xsl:attribute name="href">
            <xsl:value-of select="concat('',ItemLink)"/>
            </xsl:attribute>
            <xsl:value-of select="ItemTitle"/>
         </xsl:otherwise>
      </xsl:choose></A><xsl:apply-templates select="Item|Menu"/></LI></
xsl:otherwise></xsl:choose>
  </xsl:template>
  <xsl:template name="maketitle"><xsl:choose>
      <xsl:whentest="ItemLink=''"><xsl:value-of select="Title"/><///refuse to the select of the selec
xsl:when><xsl:otherwise><A>
      <xsl:attribute name="href"><xsl:value-of
select="concat('',Link)"/></xsl:attribute>
      <xsl:value-of select="Title"/></A></xsl:otherwise></xsl:choose>
      <UL><xsl:apply-templates select="Item"/></UL>
   </xsl:template>
   <!-- identity template -->
   <xsl:template match="@* |node()">
      <xsl:copy><xsl:apply-templates select="@* |node()"/></xsl:copy>
   </xsl:template>
</xsl:stylesheet>
```

#### 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

Below is the XSL code that is used to create TreeMenu 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:output method= "xml" version="1.0" encoding="UTF-8"

indent="yes" omit-xml-declaration= "yes"/>

<!-- Change sitepath to match you environment -->

<xsl:variable name="sitepath">/CMS400Example/</xsl:variable>

<!--

menuId must be different for each menu on the page.

This does mean that you need to use a different copy

of this file (each with their own menuId) for each

menu that you will display.
```

```
-->
<xsl:variable name="menuId">Menu1</xsl:variable>
<xsl:template match="/">
<xsl:apply-templates select="MenuDataResult/Item"/>
</xsl:template>
<xsl:template match="Item[parent::MenuDataResult]">
<xsl:apply-templates select="Item/Menu"/>
</xsl:template>
<xsl:template match="Item[child::Menu]">
<xsl:apply-templates select="Menu"/>
</xsl:template>
<xsl:template match="Menu">
<xsl:choose>
<xsl:whentest="Link=''">
<span>
<xsl:attribute name="class">handcur</xsl:attribute>
<xsl:attribute name="onclick">showBranch('<xsl:value-of</pre>
select="$menuId" />_<xsl:value-of select="ID"/>')</xsl:attribute>
<img>
<xsl:attribute name="src"><xsl:value-of select="$sitepath" /</pre>
>workarea/images/application/ico_menu-closed.gif</xsl:attribute>
<xsl:attribute name="id">I<xsl:value-of select="$menuId" /</pre>
>_<xsl:value-of select="ID"/></xsl:attribute>
</img>
<xsl:choose>
<xsl:whentest="ImageOverride='True'">
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="Image"/></</pre>
xsl:attribute>
```

```
</img>
```

```
</xsl:when>
<xsl:otherwise>
<xsl:iftest="not(Image='')">
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="Image"/></</pre>
xsl:attribute>
</img><xsl:text> </xsl:text>
</xsl:if>
<xsl:value-of select="Title"/>
</xsl:otherwise>
</xsl:choose>
</span>
<br/>br/>
</xsl:when>
<xsl:when test="not(Link)">
</xsl:when>
<xsl:otherwise>
<span>
<xsl:attribute name="class">handcur</xsl:attribute>
<xsl:attribute name="onclick">showBranch('<xsl:value-of</pre>
select="$menuId" />_<xsl:value-of select="ID"/>')</xsl:attribute>
<img>
<xsl:attribute name="src"><xsl:value-of select="$sitepath" /</pre>
>Workarea/images/application/ico_menu-closed.gif</xsl:attribute>
<xsl:attribute name="id">I<xsl:value-of select="$menuId" /</pre>
>_<xsl:value-of select="ID"/></xsl:attribute>
</img>
<xsl:choose>
<xsl:whentest="ImageOverride='True'">
```

```
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',Link)"/></xsl:attribute>
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="Image"/></</pre>
xsl:attribute>
</img>
</A>
</xsl:when>
<xsl:otherwise>
<xsl:iftest="not(Image='')">
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',Link)"/></xsl:attribute>
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="Image"/></</pre>
xsl:attribute>
</img>
</A><xsl:text> </xsl:text>
</xsl:if>
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',Link)"/></xsl:attribute>
<xsl:value-of select="Title"/>
</A>
</xsl:otherwise>
</xsl:choose>
</span>
<br/>br/>
</xsl:otherwise>
</xsl:choose>
<span class="branch">
```

```
<xsl:attribute name="id"><xsl:value-of select="$menuId" /</pre>
>_<xsl:value-of select="ID"/></xsl:attribute>
<xsl:apply-templates select="Item"/>
</span>
</xsl:template>
<xsl:template match="Item">
<xsl:choose>
<xsl:whentest="ItemLink=''">
<span class="handcur">
<xsl:choose>
<xsl:whentest="ItemImageOverride='True'">
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="ItemImage"/><///r>
xsl:attribute>
</img>
</xsl:when>
<xsl:otherwise>
<xsl:iftest="not(ItemImage='')">
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="ItemImage"/><///r>
xsl:attribute>
</img><xsl:text> </xsl:text>
</xsl:if>
<xsl:value-of select="ItemTitle"/>
</xsl:otherwise>
</xsl:choose>
</span><br/>
<xsl:apply-templates select="Item|Menu"/>
```

```
Ektron CMS400.NET Developer Reference Manual, Version 5.0, Revision 11
```

```
</xsl:when>
```

```
<span>
```

```
<xsl:attribute name="class">handcur</xsl:attribute>
```

<xsl:choose>

```
<xsl:whentest="ItemType='ExternalLink'">
```

<xsl:choose>

<xsl:whentest="ItemImageOverride='True'">

<A><xsl:attribute name="href"><xsl:value-of select="ItemLink"/></
xsl:attribute>

<img>

<xsl:attribute name="border">0</xsl:attribute>

<rsl:attribute name="src"><xsl:value-of select="ItemImage"/></ xsl:attribute>

</img>

</A>

</xsl:when>

<xsl:otherwise>

<xsl:iftest="not(ItemImage='')">

<A><xsl:attribute name="href"><xsl:value-of select="ItemLink"/></
xsl:attribute>

<img>

<xsl:attribute name="border">0</xsl:attribute>

<xsl:attribute name="src"><xsl:value-of select="ItemImage"/></
xsl:attribute>

</img>

</A><xsl:text> </xsl:text>

</xsl:if>

<A><xsl:attribute name="href"><xsl:value-of select="ItemLink"/></
xsl:attribute>

<xsl:value-of select="ItemTitle"/>

```
</A>
```

</xsl:otherwise>

</xsl:choose>

```
</xsl:when>
```

```
<xsl:whentest="ItemType='Javascript'">
```

<xsl:choose>

```
<xsl:whentest="ItemImageOverride='True'">
```

```
<A><xsl:attribute name="onClick">Javascript:<xsl:value-of
select="ItemLink"/></xsl:attribute><xsl:attribute name="href">#</
xsl:attribute>
```

<img>

```
<xsl:attribute name="border">0</xsl:attribute>
```

<xsl:attribute name="src"><xsl:value-of select="ItemImage"/></
xsl:attribute>

</img>

</A>

```
</xsl:when>
```

<xsl:otherwise>

<xsl:iftest="not(ItemImage='')">

```
<A><xsl:attribute name="onClick">Javascript:<xsl:value-of
select="ItemLink"/></xsl:attribute><xsl:attribute name="href">#</
xsl:attribute>
```

<img>

<xsl:attribute name="border">0</xsl:attribute>

<xsl:attribute name="src"><xsl:value-of select="ItemImage"/></
xsl:attribute>

</img>

</A><xsl:text> </xsl:text>

</xsl:if>

```
<A><xsl:attribute name="onClick">Javascript:<xsl:value-of
select="ItemLink"/></xsl:attribute><xsl:attribute name="href">#</
xsl:attribute>
```

<xsl:value-of select="ItemTitle"/>

```
</A>
</xsl:otherwise>
</xsl:choose>
</xsl:when>
<xsl:otherwise>
<xsl:choose>
<xsl:whentest="ItemImageOverride='True'">
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',ItemLink)"/></xsl:attribute>
<img>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="ItemImage"/><///r>
xsl:attribute>
</img>
</A>
</xsl:when>
<xsl:otherwise>
<xsl:iftest="not(ItemImage='')">
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',ItemLink)"/></xsl:attribute>
<imq>
<xsl:attribute name="border">0</xsl:attribute>
<xsl:attribute name="src"><xsl:value-of select="ItemImage"/><///i>
xsl:attribute>
</img>
</A><xsl:text> </xsl:text>
</xsl:if>
<A><xsl:attribute name="href"><xsl:value-of
select="concat('',ItemLink)"/></xsl:attribute>
<xsl:value-of select="ItemTitle"/>
</A>
```

</xsl:otherwise>

</xsl:choose>

</xsl:otherwise>

</xsl:choose></span><br/>

<xsl:apply-templates select="Item|Menu"/>

</xsl:otherwise>

</xsl:choose>

</xsl:template>

<!-- identity template -->

<xsl:template match="@\* |node()">

<xsl:copy><xsl:apply-templates select="@\* |node()"/></xsl:copy>

</xsl:template>

</xsl:stylesheet>

## **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 via a style sheet.

CMS Developer	
Book >	
News >	"All-Stars" Customer
	Web Design Firms
	Enhanced Workflow
	Visual Rapid CMS
	Ektron Partners and

Before using the DhtmlMenu Server Control, you must add items to a menu in the Ektron CMS400.NET workarea. To learn about creating menus, see "Working with Menus" in the CMS400.NET User Manual.

The DhtmlMenu Server Control also lets developers add content to a menu. When you are logged in to your **Ektron CMS400.NET** site, two menu items are added to each 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".

CMS Develop	er		
Book	>	ASP.NET Unle	eashed
News	>	Programming	C#.NET
Add		VB.NET How t	to
📝 Edit Menu		🔁 👌 Add	
Add or Edit		📝 Edit Menu	
whole menu		Add or Edit submenu	

This subsection contains the following topics:

- "Menu Server Control vs. DhtmlMenu Server Control" on page 178
- "DhtmlMenu Server Control Properties" on page 178
- "Using a Style Sheet" on page 182

#### Menu Server Control vs. DhtmlMenu Server Control

See "Menu Server Control vs. DhtmlMenu Server Control" on page 160.

### **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.Net help.

Property	Value	Data Type
Align	Specifies the horizontal alignment of the root menu text. The default is <b>Center.</b> <b>NotSet</b> - Use the default setting. <b>Left</b> - Aligns text to the left. <b>Center</b> - Aligns text to the center. <b>Right</b> - Aligns text to the right. <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 Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	String
BackColor	Sets the background color for the root menu. This property appears in the <b>Appearance</b> category.	String
CssClass	Apply a .css class to the root menu. Enter the class name from the .css defined in the Stylesheet property. <i>See Also:</i> "Stylesheet" on page 181 Since pop_style.css is the default style sheet, you can use its classes without defining it in the Stylesheet property. For example, if you enter lowtext in the property, the root menu changes as shown below.	String
	No CssClass applied     IowText CssClass applied       MenuExample     MenuExample	
DefaultMenuID	The ID of a menu 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 menu, use the CMS Explorer to browse to it. See Also: "Using CMS Explorer to Browse Your Ektron CMS400.NET Site" on page 32	Integer
DynamicParameter	Select <b>menu_id</b> . When you do, this server control uses the menu passed as a URL parameter.	String

Property	Value	Data Type
ForeColor	Sets the color of the font for the root menu. This property appears in the <b>Appearance</b> category.	
Hide	Used to hide a menu in design time and run time. <b>True</b> = Hide menu <b>False</b> = Show menu	Boolean
HorizontalDisplay	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. The default is False True = Submenu appears to the right of the root menu Our Company Notices Company Policies Company Calendar Web Forms Print Forms Latest Changes Sitemap False = Submenu appears below the root menu Root Menu Notices Company Directory Notices Company Directory Notices Company Directory Notices Company Policies Company Directory Notices Company Calendar Web Forms Print Forms Latest Changes Submenu Notices Company Calendar Web Forms Print Forms Latest Changes Sitemap	Boolean
Language	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).	Integer

Property	Value	Data Type
MenuHoverOverColor	Sets the background color for the root menu when hovering over it.	String
MenuStyle	A specified inline style that defines the root menu. For example, if you enter the following in the MenuStyle property box: font: bold 18px Arial;color:royalblue; background-color:slateblue	String
Stylesheet	Specify the location and style sheet for the submenu. For example: \Workarea\csslib\mytest.css. Leave blank to use the default. pop_style.css is the default style sheet. It is located in \webroot\CMS400Example\Workarea\csslib. For more information on style sheets, see "Using a Style Sheet" on page 182 The Stylesheet property can be used in conjunction with the CssClass property to define the root menu. See Also: "CssClass" on page 179	String
SubMenuBackColor	Sets the background color for the submenu.	String
SubMenuHoverOverColor	Sets the background color for the submenu when hovering over it.	String
SubMenultemHeight	Sets the height of each submenu item in pixels. The Default is 20 Example set to 20: CMS Developer Example set to 30: CMS Developer	Integer
SubMenuItemWidth	Sets the width of each submenu item in pixels. The Default is 150 Example set to 150: CMS Developer Example set to 130: CMS Developer	Integer

Property	Value	Data Type
SubMenuTopOffset	Sets the location of the top of the submenu. This number is in pixels. The default is 5	Integer
	Example set to 5:	
	Book > ASP.NET Unleashed News > Programming Off NET	
	VB.NET How to	
	Example set to 0 (zero):	
	Book > ASP.NET Unleashed	
	News > Programming C#.NET	
	VB.NET How to	
	Example set to -10:	
	ASP.NET Unleashed	
	News	
	VB.NET How to	
WrapTag	Allows a developer to specify a server control's tag.	String
	Span - The <span> tag is used to designate an inline</span>	
	<b>Div</b> - The <div> tag is used when you want to apply</div>	
	attributes to a block of code.	
	<b>Custom</b> - Allows you to use a custom tag.	

#### Using a Style Sheet

You can create your own style sheet, or modify the existing default one, pop\_style.css. It is located in \webroot\CMS400Example\Workarea\csslib.

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 in 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 the style sheet.

The following is a list of style sheet properties that can be modified in the  $pop_style.css$ .

Style Class	Description	Attribute	Default Value
.lowText	Text of the floating menu as displayed on the menu.	<ul> <li>color</li> <li>font</li> <li>text-align</li> <li>text-decoration</li> </ul>	<ul> <li>#000000 (Black)</li> <li>12px Arial, sans-serif</li> <li>left</li> <li>none</li> </ul>
.highText	Text of the floating menu when highlighted.	<ul> <li>color</li> <li>font</li> <li>text-align</li> <li>text-decoration</li> </ul>	<ul> <li>#000000 (Black)</li> <li>12px Arial, sans-serif</li> <li>left</li> <li>none</li> </ul>

Style Class	Description	Attribute	Default Value
.roothighText Note: This class only works when called by the CssClass property. Use the Font property in the Appearance category to define root menu text.	Text of the root menu as displayed on a Web page.	<ul> <li>color</li> <li>font</li> <li>text-align</li> <li>text-decoration</li> </ul>	<ul> <li>#000000 (Black)</li> <li>16px Arial, sans-serif</li> <li>center</li> <li>none</li> </ul>
.itemBorder	Border color and size of the menu.	• border	• 1px solid #666666
.itemBorderBlank	Separator border color and size of the menu.	• border	1px solid #CCCCDD

For more information on style sheets, visit http://www.w3.org/Style/CSS/.

## 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 185
- "MetaData Server Control Properties" on page 185
- "Using the MetaData Server Control" on page 186

#### 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 188.

#### 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.Net 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 32	String
DefaultContentID	The ID of the content block that the server control will get the metadata from. If you want to add metadata from several content blocks, set this property to zero (0) and use the DefaultItemList property to identify the content blocks. 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 32	Integer
DefaultItemList	A comma delimited list of content block IDs to get metadata from. DefaultContentID must be set to zero (0) so you can add IDs to the list. DefaultItemList [12,4,7]	Integer
DynamicParameter	Gets or sets the QueryString parameter to read a content ID dynamically.	String
Hide	Used to hide a Metadata Server Control in design time and run time. <b>True</b> = Hide Metadata Server Control <b>False</b> = Show Metadata Server Control	boolean

### 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="MetaDatal" 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>
<pre><cms:metadata defaultcontentid="12" id="Metadata2" runat="server"></cms:metadata></pre>
<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"/>
<pre><meta content="http://schemas.microsoft.com/intellisense/i&lt;/pre&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;/HEAD&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;body&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;form id=" form1"="" method="post" name="vs targetSchema" runat="server"/></pre>

- 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.



# MetaDataList Server Control

Use the MetaDataList Server Control to create lists based on Keyword Names and Keyword Values contained within the metadata of a content block. In the example below the KeyWordNames are circled in red and the KeyWordValues are circled in green.

Title: Contact Ektron		[English (U.S.)]
Content Summary Met	tadata Schedule Comment	
* = Required fields		
Title: (500 max.)		Separator Character ";"
Contact Ektron		×
486 characters left Default		
Keywords: (500 max.)		Separator Character ";"
email; support; ques	tions; ektron; contact;	
	The list can display the inforr choose, based on properties summary displayed. You can For information on adding Me Users Manual > "Adding or E	mation as a list of hyperlinks. You can s you set, if you would like to have a a also choose how to order the display etadata to a content block, see the Editing Metadata".

This subsection contains the following topics:

- "MetaData Server Control vs. MetaDataList Server Control" on page 189
- "MetaDataList Server Control Properties" on page 189

## 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 MetaData Server Control, see "MetaData Server Control" on page 185

### 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.Net 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 32	String
ContentType	<ul> <li>One of the following</li> <li>all types</li> <li>archived content</li> <li>archived forms</li> <li>content</li> <li>forms</li> <li>To learn about archived content, see the Ektron CMS400.NET User Manual, "Scheduling Content to Begin and End" &gt; "Setting Archive Options."</li> </ul>	String

Property	Value	Data Type
DisplayXslt	Determines how the information on the page is displayed <b>None</b> -databind only	String
	ecmNavigation - lists the title of every content block in the folder	
	See Also: "Example of ecmNavigation Display" on page 121	
	<b>ecmTeaser</b> - lists the title of every content block in the folder plus the content summary.	
	See Also: "Example of ecmTeaser Display" on page 122	
	Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page	
ExactPhrase	Determines whether the KeyWordValue needs to match the phrase exactly.	Boolean
	For example, if "site" is the KeyWordValue and the title of a content block is "Welcome to the site" and ExactPhrase is set to true, you would not see the content block in the metadata list. This is because "site" does not equal "Welcome to the site".	
	True = Match the exact phrase	
	<b>False</b> = Doesn't need to match exact phrase	
FolderID	The root folder content will be pulled from.	Integer
GetHtml	Set to <b>True</b> 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 DataGrid.	Boolean
Hide	Used to hide a metadata list in design time and run time.	Boolean
	True = Hide metadata list	
	False = Show metadata list	
Property	Value	Data Type
--------------	---	-----------
KeyWordName	The KeyWordName is the container where the KeyWordValues are located. Some examples of a KeyWordName might be "Keywords or Title" For information on creating KeyWordNames for Metadata, see the <b>Ektron CMS400.NET</b> Administrators Manual > "Managing Content" > "Metadata" > "Managing Metadata Definitions" > "Adding a Metadata Definitions"	String
KeyWordValue	The value that is associated with the KeyWordName Some examples of a KeyWordValue might be "Home; page; or company"	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
MaxNumber	The maximum number of results returned for display. <b>0</b> = unlimited	Integer
OrderBy	<ul> <li>The order of the list to be returned.</li> <li>Title - The title of the content block</li> <li>ID - The content block ID number</li> <li>Date Created - The date the content block was created</li> <li>Date Modified - The date the content block was last modified</li> <li>LastEditorLname - The last editor's last name</li> <li>LastEditorFname - The last editor's first name</li> </ul>	String
Recursive	Whether to search sub-folders of the identified root folder. The root folder is identified in the Folderpath property.	Boolean
SortOrder	Ascending or Descending	String

Property	Value	Data Type
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.</div></span>	String

# **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
- 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.microsoft.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 195

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 194
- "Using the RssAggregator Server Control" on page 195

## **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.Net 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 32	String
DisplayXslt	Determines how the information on the page is displayed	String
	None-databind only	
	ecmNavigation - lists the title of every RSS feed item	
	See Also: "Example of ecmNavigation Display" on page 121	
	ecmTeaser - lists a title and a description of every RSS feed item	
	See Also: "Example of ecmTeaser Display" on page 122	
	<b>Path to Custom Xslt</b> - Enter the path to an Xslt that determines the display of the page	
Hide	Used to hide a RssAggregator Server Control in design time and run time.	boolean
	True = Hide Metadata Server Control	
	False = Show Metadata Server Control	
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
MaxResults	The Maximum number of items from an RSS feed that are returned (0=unlimited).	Integer
URL	The RSS feed path for the server control. For example: http://msdn.microsoft.com/rss.xml	String

Property	Value	Data Type
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.</span>	String
	<ul> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> <li>Custom - Allows you to use a custom tag.</li> </ul>	

## Using the RssAggregator Server Control

These steps show how to use the RssAggregator Server Control.

- 1. Drag a 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 194

3. Build the Web form.

4.	Browse to	the Web	page in	your browser	
----	-----------	---------	---------	--------------	--

Address an http://localhost/CMS400Example/webform.aspx	× 🖻	Go
Elinks 🙆 EktronNet Welcome 💰 Ektron, Inc. 💰 CMS400 Example 💰 Welcome to RC International	al	»
<u>iTunes Top 10 Songs</u> iTunes Music Store: Today's Top 10 Songs		<
<u>1. These Boots Are Made for Walkin' (Radio Edit) - Jessica Simpson</u> These Boots Are Made for Walkin' (Radio Edit) by Jessica Simpson from the album Boots Are Made for Walkin' - Single	These	III
<u>2. Pon de Replay (Radio Edit) - Rihanna</u> Pon de Replay (Radio Edit) by Rihanna from the album Pon de Replay - Single		
3. Don't Cha (featuring Busta Rhymes) [Radio Edit] - The Pussycat Dolls & Busta R Don't Cha (featuring Busta Rhymes) [Radio Edit] by The Pussycat Dolls & Busta Rh from the album Don't Cha	<u>hymes</u> lymes	
<u>4. Best of You - Foo Fighters</u> Best of You by Foo Fighters from the album In Your Honor (Bonus Video Version)		
5 Don't Phunk With My Heart - Black Eved Peac 5. The page appears with an RSS list of information.		~

# **BreadCrumb Server Control**

The BreadCrumb Server Control creates a breadcrumb trail of visited pages. Your site visitors can use the 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.

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

Breadcrumb Trail Type	Example
Vertical with images	DEV > PR > FAQ > Search > Meta
Vertical with images and Alt text	DEV > PR > FAQ > Search > Meta

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 199
- "Types of BreadCrumb Server Controls" on page 199
- "BreadCrumb Server Control Properties" on page 201
- "Using the BreadCrumb Server Control" on page 203
- •

# 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 Controls

BreadCrumb Server Control properties are divided into two categories:

- "Properties that Determine a Breadcrumb Trail's Appearance" on page 199
- "Properties that Determine How Pages Associated with the Form Appear on a Breadcrumb Trail" on page 200

### **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 201
DisplayStyle	Whether it appears horizontally or vertically	"DisplayStyle" on page 201
LinkLastItem	Whether the last item is a hyperlink	"LinkLastItem" on page 202
LinkTarget	The type of window that appears when a user clicks an item	"LinkTarget" on page 203
MaxItems	The maximum number of items	"MaxItems" on page 203
Mode	whether the breadcrumb trail appears as hyperlinked text or plain text.	"Mode" on page 203
Separator	Symbols or characters used to separate items	"Separator" on page 203

# 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 202
IconAlt	"Alt" text associated with image specified in IconPath property	"IconAlt" on page 202
IconPath	Image to define it	"IconPath" on page 202

## **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.Net help.

Property	Response	Data Type
CurrentPageIndicator	If desired, enter one or more symbols or characters that identify the current page in the breadcrumb trail. See example below.	String
	Dev > WebForm > pr > FAQ > Meta*	
	These characters appear <i>after</i> the image or title that identifies the current page in the breadcrumb trail.	
DisplayStyle	Indicate how to display the breadcrumb trail: horizontally or vertically. See Also: "Breadcrumb Trail Type" on page 197	String

Property	Response	Data Type
DisplayTitle	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> . <i>See Also:</i> "BreadCrumb Properties Apply to Web Forms, Not Web Pages" on page 199 The default is <b>Title</b> . If you define an image in the <b>IconPath</b> property below, the image appears in the trail, followed by this text.	String
IconAlt	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.	String
IconPath	If you want the breadcrumb trail to display an image to identify this Web form, enter the path to the image. Important: The image location must be relative to the Web root. For example: \CMS400Example\Workarea\Images\bc_meta_icon.gif On the breadcrumb trail, the image precedes any text defined in the <b>DisplayTitle</b> property.	String
LinkLastItem	Use this property to determine whether the last breadcrumb item appears as a hyperlink on this Web form. If this property is set to <b>true</b> , and a user clicks the item, the current page reappears. <b>True</b> = last item is a hyperlink <b>False</b> = last item is an image and/or text only; the user cannot click on it	Boolean

Property	Response	Data Type
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 <b>_self</b> .	String
	_ <b>Self -</b> opens in same window	
	_ <b>Top</b> - opens in parent window	
	_Blank - opens in new window	
	_ <b>Parent</b> - opens in the parent frame	
MaxItems	Enter the maximum number of items in the breadcrumb trail on this Web form.	Integer
	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.	
	To allows an unlimited number of breadcrumb trail items on this Web form, enter zero ( <b>0</b> ).	
Mode	Allows you to make the breadcrumb trail appear as non-	String
	<b>Normal</b> = breadcrumb trail is hyperlinked	
	PR > Webform > <u>Search</u> > Meta	
	<b>DisplayOnly</b> = breadcrumb trail is plain text	
	PR > Webform > Search > Meta	
Separator	Enter one or more characters to separate the items in a breadcrumb trail on this Web form.	String
	The default character is the greater than sign (>).	
	Dev > WebForm	

## 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.

- 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 202.
- **NOTE** To customize the look of your breadcrumb trail, configure the other properties. For more information, see"BreadCrumb Server Control Properties" on page 201.
  - 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: BreadCrumbl.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.

# **Dreamweaver Extension Support**

Ektron CMS400.NET has a custom Macromedia<sup>®</sup> Extension Package, which allows you to not only create your templates in Dreamweaver<sup>®</sup> or UltraDev<sup>TM</sup>, but also to insert the custom functions through Dreamweaver<sup>®</sup>.

To do this, all you need is Ektron CMS400.NET, Dreamweaver®/ UltraDev<sup>TM</sup>, 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 207
- "Preparing Dreamweaver for Ektron CMS400.NET Functions" on page 208
- "Installing the Extensions" on page 208
- "Setup" on page 213
- "Using the Ektron CMS400.NET Extensions" on page 217
- "Inserting Custom Functions" on page 226

### 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 MX 2004\Configuration\Translators\ASP.NET\_VB\

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 MX 2004\Configuration\Translators\ASP.NET\_Csharp\

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 208

### 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:

Macrom	nedia Extension Manager
!	An older copy of the file 'C:\Documents and Settings\shughes1\Application Data\Macromedia\Dreamweaver MX 2004\configuration\ThirdPartyTags\EKTRONLogin.gif' already exists. Do you want to replace it?
	Yes Yes to All No Cancel

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<sup>®</sup>, 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<sup>®</sup> Manual.

To install an extension, follow these steps.

 Begin by following the path in Dreamweaver: Commands > Manage Extensions...



#### 2. The Macromedia Extension Manager is displayed.

🚳 Macromedia Extension Manager				
<u>File</u> <u>H</u> el	þ			
2	Treamweaver MX	2004	• 🔺	
On/Off	Installed Extensions	Version	Туре	Author
I				~

3. From this dialog box, follow the path:

#### File > Install Extension...

4. The Select Extension to Install window opens.

Select Extension to Install	×
Look in: 🔁 Extension Manager 🔹 🗲 🗈 📸 📰 🗸	
Downloaded Extensions	
Samples	
File name: Install	
Files of type:         Macromedia Extension Packages (*.mxp)         Cancel	

5. Locate the Macromedia<sup>®</sup> Extension Package (.mxp) file that corresponds to Ektron CMS400.NET. By default, the file gets installed to the following directory:

Select Extension to Install	? 🔀
Look in: 🔁 Utilities	▪ ← 🗈 📸 🎟 -
C400_ASP_PHP_CF_JSP.mxp	
File <u>n</u> ame:	Install
Files of type: Macromedia Extension Packages (*.n	mxp) 💌 Cancel

- 6. After you select the file, click the **Install** button.
- 7. The installation begins with the Macromedia<sup>®</sup> Extensions Disclaimer. Click **Accept**.

8. The extension begins to install. Once installed, you see a confirmation box.

Macrom	edia Extension Manager
1	The 'EKTRON CMS400' extension has successfully been installed. In order for the changes to take effect, you must close and then restart Dreamweaver MX 2004.
	OK

- 9. Click **OK** to continue.
- 10. The Macromedia Extension Manager is again displayed, but with the added extension.

Macromedia Extension Manager	
<u>File H</u> elp	
🍰 📋 Dreamweaver MX 2004 💽 💽	
On/Off Installed Extensions Version Type	Author
🗹 🛛 🦂 EKTRON CMS400 (ASP CF JSP PHP) 5.0.5 Command	Ektron, Inc.
🔽 🥔 EKTRON CMS400.NET (.NET Server controls) 5.0.5 Command	Ektron, Inc.
Insert and update Ektron CMS400 Server controls into an ASP.NET Web page using Dre or Dreamweaver MX 2004. With an Ektron CMS400-enabled Web site, business users can author, publish and mana content (e.g., text, images, tables, files, hyperlinks), while ensuring that necessary parame place for maintaining control over the site's look and feel, navigation, and security. Non-technical content contributors can log into the Ektron CMS400 application directly fr site. They check out a "content block" (content from a specific page or section of a pag new content block, then usea familiar WYSIWYG editor to format text; copy and paste frr Windows applications without losing formatting; insert tables, images, and hyperlinks; che and more. Content is published with a few simple clicks of the mouse.	eamweaver
11. If needed, exit Dreamweaver <sup>®</sup> , 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 217 for more information on using Dreamweaver® to create and edit templates for Ektron CMS400.NET.

### Setup

Before you can use Dreamweaver®/UltraDev<sup>TM</sup> 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
- Work Online
- 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...

٨	lacro	medi	a Drean	nweave	r MX	2004					
e	Edit	View	Insert	Modify	<u>T</u> ext	<u>C</u> ommands	Site	<u>W</u> indow	Help		
• 69	Inse	rt	Common	Layou	t   For	Start <u>R</u> ec Play <u>R</u> ec Edit Com <u>G</u> et More <u>M</u> anage B	ording orded C mand Li Comm Extensi	Command ist ands ons		Ctrl+Shift+X	ites
						Apply So Apply So Clean Up Clean Up Add/Rem	urce Fo urce Fo HTML. Word H	ormatting ormatting i  HTML etscape Re	to Selection esize Fix		4
						<u>O</u> ptimize <u>C</u> reate W Se <u>t</u> Color	Image /eb Pho Schem	in Firewor to Album ie	ks		-
						<u>F</u> ormat T <u>S</u> ort Tabl	able e				ion
						Ektron Cl Ektron Cl	MS400- MS400.	(ASP CF ) NET (Serv	ISP PHP) ver Controls)		Setup CMS Function

2. The Setup dialog box is displayed on the screen.

Ektron CMS400.NET (Server Controls) Setup

Ektron CMS	400 Setup	X
Username:	admin	Update Setup
Password:	•••••	Cancel
Domain:		Help
Server File:	http://localhost/cms400example/workarea/dreamweaver.aspx	
Include:	<%@ Register TagPrefix="cms" Namespace="Ektron.Cms.Controls" Assemb	

	Ektron CMS400 (ASP CF JSP PHP) Setup	
Ektron CMS	400-Platform Setup	
Username:	admin	Update Setup Cancel
Password:	•••••	Help
Domain:		<u> </u>
Server File:	http://localhost/cms400Example/workarea/dreamweaver.aspx	
Server:	⊙ ASP ○ JSP ○ CF ○ PHP	
Work O	nline (check is default)	
Include:	#include file="site_scripts_path.asp"	
Check f	or include (check is default)	

3. 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 "Use the following table to help you update your setup information. This table refers to Ektron CMS400 - (ASP CF JSP PHP) Setup." on page 216

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.

Use the following table to help you update your setup information. This table refers to Ektron CMS400 - (ASP CF JSP PHP) Setup.

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	Choose the language of the server.
	ASP - for Active Server Pages
	• <b>CFM</b> - for ColdFusion pages
	JSP - Java Server Pages
	PHP - Hypertext Preprocessor
	Note: If you want to use a different language, you need to run this setup again and choose a different language.
Work Online	By checking this box, you will have access to the database to obtain the correct folder and content information. For example, when adding a content tag, you are able to see the content titles in a drop down list. This makes it easy to insert the right content tag. Checked is the default.
	Checked - work offline
	Unchecked - cannot work offline
Include	Each template requires an include file to point to the API with the definitions for the custom functions. Here, specify the include file.
	• For ASP use: #include file="site_scripts_path.asp"
	For CFM use: Leave blank
	<ul> <li>For JSP use: &lt;%@ include file="applicationapi.jsp" %&gt;</li> </ul>
	<ul> <li>For JSP use: <?php include "site_scripts_path.php"; ?></li> </ul>

Field	Description
Check for Include	If checked the Include statement is added to your page. Checked is the default.
	Checked - include statement added to your page
	Unchecked - include statement is not added to your page

- 4. Click Update Setup.
- 5. A message appears.

Macromedia Dreamweaver MX 2004	$\times$
Successful connection to the dreamweaver server file "http://localhost/cms400example/workarea/dreamweaver.aspx" has been made and the username and password are correct. Working Online will log you out of the CMS400 application.	
ОК	

**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.

### Using the Ektron CMS400.NET Extensions

Now that the Macromedia<sup>®</sup> Extension Package has been installed for Dreamweaver, you can create and edit templates for Ektron CMS400.NET.

To explain how to use this, we use templates that are in the Ektron CMS400.NET installation, but with the custom functions stripped out.

**NOTE** You can also create your own templates instead of using the ones in the Ektron CMS400.NET installation.

The following subsections are contained in this section:

- "Custom Function Symbols" on page 219
- "Ektron CMS400.NET Command Menus" on page 221

In Dreamweaver, begin by opening the file:

🥑 Macromedia Dreamweaver MX 2004 - [loginexample (CMS400Example/log				
<u>File Edit View Insert M</u> odify <u>T</u> ext <u>C</u> ommands <u>S</u> ite <u>W</u> indow <u>H</u> elp				
Common 🔻 🔌 🖃 🌡 🗒 🗸 🧔 🗸 📅 🖳 🖹 🗸 🗊				
loginexample.aspx* _ ₽ ×				
Code 📇 Split 📴 Design 📝 Title: loginexample 🖓 👫				
10 <meta content="http://schemas.&lt;/pre&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;12 &lt;LINK href=" default.css"="" form1"="" method="post" name="vs_defaultClientScript" rel="styles}&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;13 &lt;/HEAD&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;14 &lt;body&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;15 &lt;form id=" runat="server" type="text/css" vs_targetschema"=""/>				
16				
17				
18				
19				
20 <p> </p>				
21 <table border="1" cellpadding="0" cel⊻<="" td="" width="95%"></table>				
A				
Description				
Files: • loginexample.aspx				
• loginexample.vb				
<body> <form#form1>  470 × 164 - 6K / 1 sec</form#form1></body>				

\webroot\CMS400Example\loginexample.aspx

**NOTE** The custom function that is inserted into this template by default has been stripped out.

### **Custom Function Symbols**

During the process of inserting functions, symbols are used to represent the custom display functions. The following table explains each symbol.

Symbol	Represents	More Information
💰 Login	Login/Logout button	"Login" on page 227
ektron Content Block	Static content block.	"Inserting a Content Block" on page 230
ektron Content Block Dynamic URL pagename?id=xx	Dynamic content block	"Inserting a Dynamic Content Block" on page 234
ektron Collection	Collection	"Inserting a Collection" on page 236
ektron List Summary	List Summary	"Inserting a List Summary" on page 240
ektron Search Tag	Search	"Inserting a Search Function" on page 251
ektron Search Display	Search display function	"Inserting a Search Display Function" on page 257
No symbol appears when Metadata is inserted in a Web page.	Metadata	"Inserting a Metadata Function for ASP.NET pages" on page 245

Symbol	Represents	More Information
ektron Single Summary	Single Summary	"Inserting a Single Summary Function" on page 259
ektron	Random Content	"Inserting a Random Content Function" on page 262
ektron Random Summary	Random Summary	"Inserting a Random Summary Function" on page 264
ektron Calendar	Event Calendar	"Inserting an Event Calendar Function" on page 266
ektron Form Block	Form Content Block	"Inserting a Form" on page 269
ektron XML Index Search Tag	XML Index Search	"Inserting an XML Index Search" on page 276
ektron display by Metadata	Display By Meta Value	"Inserting Display By Meta Value" on page 282
ektron Content Block	Multipurpose content block.	"Inserting a Multipurpose content block" on page 273
	Advanced Calendar Tag	"Inserting an Advanced Calendar Tag" on page 274

Symbol	Represents	More Information	
ektron Menu	Standard Menu	"Inserting a Standard Menu" on page 287	
ektron Menu	DHTML Menu	"Inserting a DHTML Menu" on page 290	
Language: English (U.S.) ✓	Language Select	"Inserting a Language Select Box" on page 296	
ektron Lang-API	Language API	"Inserting a Language API" on page 298	
RSS	RSS Aggregator	"Inserting an RSS Aggregator" on page 301	

### **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) > CMS Function...
- Commands > Ektron CMS400 (ASP CF JSP PHP) > CMS Function...

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) > CMS Function...

If you are using ASP, ColdFusion, JSP or PHP pages, click

Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...

3. The Ektron CMS400.NET commands menu is displayed.



#### If you use ASP.NET pages:



### If you use ASP, CF, JSP or PHP pages:

4. Click on the command for the task you would like to perform. The following table explains each command.

Command	Description	ASP.NET	ASP - CF JSP - PHP	More Information
Setup	Opens Ektron CMS400.NET Setup information dialog for Dreamweaver.	$\checkmark$	$\checkmark$	"Setup" on page 213
CMS Function				
Login	Inserts an Ektron CMS400.NET login function into the template.	$\checkmark$	$\checkmark$	"Login" on page 227

Command	Description	ASP.NET	ASP - CF JSP - PHP	More Information
Content	Inserts an Ektron CMS400.NET content block function into the template.	$\checkmark$	$\checkmark$	"Inserting a Content Block" on page 230
Dynamic	Inserts an Ektron CMS400.NET dynamic content block placeholder function into the template.		<b>~</b>	"Inserting a Dynamic Content Block" on page 234
Collection	Inserts an Ektron CMS400.NET collection function into the template.	$\checkmark$	$\checkmark$	"Inserting a Collection" on page 236
List Summary	Inserts an Ektron CMS400.NET list summary function into the template.	$\checkmark$	$\checkmark$	"Inserting a List Summary" on page 240
Metadata	Inserts an Ektron CMS400.NET metadata function into the template.	~	~	"Inserting a Metadata Function for ASP.NET pages" on page 245
Search	Inserts an Ektron CMS400.NET search function into the template.	$\checkmark$	$\checkmark$	"Inserting a Search Function" on page 251
Search Display Tag	Inserts an Ektron CMS400.NET search display function into the template.		<b>~</b>	"Inserting a Search Display Function" on page 257
Single Summary Tag	Inserts an Ektron CMS400.NET single summary function into the template.		<b>~</b>	"Inserting a Single Summary Function" on page 259
Random Content Tag	Inserts an Ektron CMS400.NET random content function into the template.		$\checkmark$	"Inserting a Random Content Function" on page 262

Command	Description	ASP.NET	ASP - CF JSP - PHP	More Information
Random Summary Tag	Inserts an Ektron CMS400.NET random summary function into the template.		~	"Inserting a Random Summary Function" on page 264
Calendar	Inserts an Ektron CMS400.NET event calendar function into the template.	<b>~</b>	<b>~</b>	"Inserting an Event Calendar Function" on page 266
Form	Inserts an Ektron CMS400.NET form content block into the template.	$\checkmark$	$\checkmark$	"Inserting a Form" on page 269
Multipurpose Tag	Insert a content block or form content block into the template.		~	"Inserting a Multipurpose content block" on page 273
Advance Calendar Tag	Inserts a calendar into the template.		<b>~</b>	"Inserting an Advanced Calendar Tag" on page 274
XML Index Search	Inserts an Ektron CMS400.NET XML Index Search into the template.	<b>~</b>		"Inserting an XML Index Search" on page 276
Display By Meta Value	Inserts an Ektron CMS400.NET display by meta value block into the template.	~		"Inserting Display By Meta Value" on page 282
Standard Menu	Inserts an Ektron CMS400.NET standard menu into the template.	$\checkmark$		"Inserting a Standard Menu" on page 287
DHTML Menu	Inserts an Ektron CMS400.NET DHTML menu into the template.	$\checkmark$		"Inserting a DHTML Menu" on page 290
Language Select	Inserts an Ektron CMS400.NET language select box into the template.	$\checkmark$		"Inserting a Language Select Box" on page 296

Command	Description	ASP.NET	ASP - CF JSP - PHP	More Information	
Language API	Inserts an Ektron CMS400.NET Language API into the template.	$\checkmark$		"Inserting a Language API" on page 298	
RssAggregato r	Aggregates an RSS feed from a RSS Web site.	<b>~</b>		"Inserting an RSS Aggregator" on page 301	
XML Function	XML Function				
Content XSLT Tag	Inserts an XML content block with your choice of XSLT in the template. NOTE: Used to insert an XML content block with an XSLT other than the default applied to it.		~	"Insert Content XSLT Tag" on page 304	

## **Inserting Custom Functions**

### **CMS** Function

The CMS Function commands let you insert standard Ektron CMS400.NET display functions into your template.

### **Accessing CMS Content Functions**

To access the CMS Content commands menu, follow one of these paths:

• Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...

or

 Commands > Ektron CMS400.NET (Server Controls) > CMS Function...


A list of all CMS Content functions that can be inserted is displayed.

The following sections explain how to insert each function.

### Login

For a description of the Login button, see "Login Server Control" on page 63.

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 Function...

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.

Insert Ektron CMS400 Login Ta	g	×
Paints the login or logout button which use the CMS400 application. ID Hide: Visible: (Positioning info - GridLayout) Style: Wrap tag:	h allows the user to login and  Cancel  Login 1  Help  Leave blank if none Span	

**IMPORTANT!** When inserting a login function using ASP, CF, JSP or PHP, the dialog box that appears has no settings.

Ektron CMS400 Login Tag	
Paints the login or logout button which allows the user to login and use the CMS400 application.	Insert Cancel Help

4. Use the following table to assist you with inserting a Login button.

Field	Description
ID	A unique name for the function you are inserting.
Used in ASP.NET only.	

Field	Description
Hide Used in ASP.NET only.	Used to hide a Login button at run time.  Checked = Hide Login button Unchecked = Show Login button Note: If Hide and Visible are both checked, the property is hidden.
Visible Used in ASP.NET only.	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout) 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 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.</div></span>

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	<cms:login id="Login1" runat="server"></cms:login>
ASP	<% ecmlogin %>
ColdFusion	<cfmodule template="#request.ecm.AppPath#ecmLogin.cfm"></cfmodule>
JSP	<%= ecmLogin() %>
PHP	php ecmlogin() ?

# **Inserting a 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) > CMS Function...

For a description of the Content Block function, see "ContentBlock Server Control" on page 70.

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

Ektron CMS400 Content control	
Displays and manages a content block. Control ID: ContentBlock1 Content ID:	Insert Cancel Help
Select the category to retrieve list of content	
Select the content block to insert into the page 💙	
Dynamic Content Parameter: None 💌	
Hide:	
Visible:	
(Positioning info - GridLayout) Style: leave blank if none	
Wrap tag: Span 🗸	

4. Use the following table to help you insert a content block.

Field	Description
Control ID Used in ASP.NET only.	A unique name for the function you are inserting.

Field	Description
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 Used in ASP.NET only.	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. None: Displays a static content block. id: Content block is passed as a URL parameter. Important: 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 "Inserting a Dynamic Content Block" on page 234
Hide Used in ASP.NET only.	Used to hide a content block in design time and run time. <b>True</b> = Hide content block <b>False</b> = Show content block Note: If Hide and Visible are both checked, the property is hidden.
Visible Used in ASP.NET only.	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout) 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"

Field	Description
Wrap Tag 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.</div></span>

**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.

Common 🔻 🔌 🖾 🗉 🖉 🗸 📅 🖳 🖺 🖛 🐺
ContentBlock.aspx*
🐼 Code 🚍 Split 📴 Design 📝 Title: ContentBlock 🖓 🎝 🖏 🖓 🖓
15 <p> </p>
16 <pre><cms:contentblock defaultcontentid="12" id="ContentBlock1" runat="server"></cms:contentblock></pre>
17
18 Description Below
ektron
Content Block
Description
•

Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<cms:contentblock defaultcontentid="12" id="ContentBlock1" runat="server"><!--<br-->cms:contentblock&gt;</cms:contentblock>
ASP	<% ecmContentBlock(1) %>
ColdFusion	<cfmodule id="1" template="#request.ecm.AppPath#ecmContentBlock.cfm"></cfmodule>
JSP	<%= ecmContentBlock(12) %>

Platform	Code View
PHP	php ecmContentBlock(1); ?

# **Inserting a 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 324

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 **Insert Dynamic Tag...**
- 3. The Insert Ektron Dynamic Tag box is displayed

CMS400 Dynamic Content Tag	X
This tag can be used where the Content ID is being passed in as a URL parameter. The Default Content ID will be used if an ID is not passed through the URL. Insert Default Content: <a></a> Default Content ID:	Insert Cancel Help
Select the category to retrieve list of content	
Select the content block to insert into the page	

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, check off 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.

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 "Inserting a Content Block" on page 230

Platform	Code View
ASP	<% if request.QueryString("id")<> "" then ecmContentBlock(request.QueryString("id")) else ecmContentBlock(1) end if %>
ColdFusion	<cfif isdefined("url.id")=""> <cfmodule <br="" id="#url.id#">template="#request.ecm.AppPath#ecmContentBlock.cfm"&gt; <cfelse> <cfelse> <cfmodule <br="" id="1">template="#request.ecm.AppPath#ecmContentBlock.cfm"&gt; </cfmodule></cfelse></cfelse></cfmodule></cfif>
JSP	Inserting a Dynamic Content Block using Ektron's Dreamweaver Extensions is not supported in JSP.
PHP	php \$id =GetURL_Id(1); ecmContentBlock(\$id); ?

# **Inserting a Collection**

For a description of the Collection function, see "Collection Server Control" on page 119

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 Function...

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**.

Insert Ektron CMS400 Collection Tag			
Collection control properties Control ID:	Collection 1		Insert Cancel Help
Select the collec	tion to insert into the name		
Select the collect	uon to insert into the page		
Display Type:	ecmTeaser 💌		
Dynamic Content Parameter:	None 🖌		
Dynamic Collection Parameter:	None 🖌		
Random Output:	None 💌		
Hide:			
Visible:			
(Positioning info - GridLayout) Style:		leave blank if none	
Wrap tag:	Span 🗸		

3. The Insert Ektron Collection box is displayed.

**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:	Insert Cancel Help
Select the display function or fill it in above	

Field	Description
Control ID	A unique name for the function you are inserting.
Used in ASP.NET only.	
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.
Display Type	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 collection <i>See Also:</i> "Example of ecmNavigation Display" on page 121 <b>ecmTeaser</b> - lists the title of every content block in the collection plus the content summary <i>See Also:</i> "Example of ecmTeaser Display" on page 122
Dynamic Content Parameter 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.
	To make this collection dynamic, select <b>coll_id</b> . When you do, the function uses the collection passed as a URL parameter.
Used in ASP.NET only.	
Random Output	Displays random content within a collection. <b>None</b> - Displays all content in collection.
Used in ASP.NET only.	Random Teaser - Displays a random content block's teaser. Random Body - Displays a random content block's body.

### 4. Use the following table to assist you with inserting a Collection.

Field	Description
Hide Used in ASP.NET only.	Used to hide a collection in design time and run time.  Checked = Hide collection Unchecked = Show collection Note: If Hide and Visible are both checked, the property is hidden.
Visible Used in ASP.NET only.	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout) 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 Used in ASP.NET only.	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> </ul>

A graphic defines where the collection appears on your template.

	🐼 Code 🔄 Split 📴 Design 📝 Title: Collection 🖓 👯 🌚 🧭 🕅			
1	S <body></body>			
1	<pre>4 <form id="Form1" method="post" runat="server"></form></pre>			
	5 <8>			
	6 <cms:collection defaultcollectionid="6" displayxslt="ecmTeaser" id="Collection1" runat="server"></cms:collection>			
	7			
	8 <p> :</p>			
1				
1				
	ektron Collection			

Ektron CMS400.NET Developer Reference Manual, Version 5.0, Revision 11

Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<cms:collection <br="" defaultcollectionid="0" id="Collection1">DisplayXslt="ecmTeaser" runat="server"&gt;</cms:collection>
ASP	<% ecmCollection 1,"ecmNavigation" %>
ColdFusion	<cfmodule <br="" displayfunction="ecmNavigation" id="1">template="#request.ecm.AppPath#ecmCollection.cfm"&gt;</cfmodule>
JSP	<%= ecmCollection(6,"ecmTeaser") %>
PHP	php ecmCollection(1,"ecmNavigation"); ?

### **Inserting a List Summary**

For a description of the List Summary function, see "ListSummary Server Control" on page 113.

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 Function...

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.

Ektron CMS400 List Summary (	Control	
ListSummary is used to display a list o	Insert	
Control ID:	ListSummary1	Cancel
Folder:		Help
Sele	ct the folder or fill it in above	
Recursive:		
Order By:	Title 🖌 Ascending 🗸	
Max Number of results:	0 (0- is unlimited)	
Display Type:	ecmTeaser 💙	
Dynamic Content Parameter:	None 💙	
Random Output:	None	
Content Type:	Content	
Hide:		
Visible:		
(Positioning info - GridLayout) Style:	leave blank if none	
Wrap tag:	Span Y	

3. The Insert Ektron List Summary box is displayed.

IMPORTANT!	When inserting a list summary using ASP, CF, JSP or PHP, a different dialog box appears.
	Ektron CMS400 List Summary Tag
	Used to display a list of content block summaries. Folder: Cancel
	Recursive:
	Show Summary: 🔽
	Style Info:
	Order By: Title 💙 asc 🗸
	Show Info: Date Modified 🐱
	Max Number of Summaries: 0

4. Use the following table to assist you with inserting a metadata control.

Field	Description
Control ID	A unique name for the function you are inserting.
Used in ASP.NET only.	
Folder	Enter the folder name manually. The default is the root or "\". 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
Order By	Specify how the summaries are ordered. Sort by the following criteria.
	• Title - The title of the content block
	• Date Created - The date the content block was created
	• Date Modified - The date the content block was last modified
	Last Author Last Name - The last author's last name
	Last Author First Name - The last author's first name
	Ascending - Alphabetic or numerical order
	• Descending - Reverse alphabetic or numerical order
Max Number of results	Enter a number to specify how many results are displayed on the page (0=unlimited).
Display Type	Determines how the information on the page is displayed.
Used in ASP.NET only.	ecmNavigation - lists the title of every content block in the folder
	See Also: "Example of ecmNavigation Display" on page 121 ecmTeaser - lists the title of every content block in the folder plus the content summary See Also: "Example of ecmTeaser Display" on page 122
Dynamic Content Parameter	Checks the QueryString for this value and replaces the list summary with a content block when specified. Choose <b>None</b> to always display
Used in ASP.NET only.	List Summary.
Random Output	Displays random content within a list summary.
Used in ASP.NET only.	<b>None</b> - Displays all content in list summary.
	Random Body - Displays a random content block's body.
Content Type	Defines what type of list summary appears.
Used in ASP.NET only.	Content - Displays only content blocks in the folder.
·	Forms - Displays only forms in the folder.
	Archive_Content - Displays archived content from the folder. Archive_Forms - Displays archived forms from the folder.

Field	Description
Hide Used in ASP.NET only.	Used to hide a list summary in design time and run time.  Checked = Hide list summary  Unchecked = Show list summary  Note: If Hide and Visible are both checked, the property is hidden.
Visible Used in ASP.NET only.	Used to show or hide the function at run time.  Checked = Hide function  Unchecked = Show function  Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout) 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 Used in ASP.NET only.	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is <b>Span</b>.</li> <li><b>Span</b> - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li><b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.</div></li> </ul>

A graphic defines where the list summary appears on your template.

pr.aspx*
🛛 🐼 Code 📇 Split 📴 Design 📝 Title: pr
13 <form 1d="form1" method="post" runat="server"></form>
14 <2>
15 (CMS:ListSummary 1-"ListSummary1" runat="server" FolderPath="\" OrderKey="DateCreated" Direction="Descending"
16 DisplayType="ecmieaser" DisplayXsit="ecmieaser" ContentParameter="1d">
10 CIT Description HS
ektron List Summary
Description

Shown below is an exan	ple of the code that is inserted:
------------------------	-----------------------------------

Platform	Code View
.NET	<cms:listsummary <br="" folderpath="\" id="ListSummary1" runat="server">OrderKey="DateCreated" Direction="Descending" DisplayType="ecmTeaser" DisplayXslt="ecmTeaser" ContentParameter="id"&gt;</cms:listsummary>
ASP	<% ecmListSummary "\Human Resources",1,1,"","DateModified,desc", "DateCreated",0,"","" %>
ColdFusion	<cfmodule <br="" folder="\Human&lt;br&gt;Resources" maxnumber="0" recursive="1" showsummary="1" styleinfo="" template="#request.ecm.AppPath#ecmListSummary.cfm">OrderBy="DateModified,desc" ShowInfo="DateCreated"&gt;</cfmodule>
JSP	<%= ecmListSummary("\Human Resources",0,1,"", "Title,asc","DateModified",0,"","") %>
PHP	php ecmListSummary("\Human Resources",1,1,"",<br "DateModified,desc","DateCreated",0,"",""); ?>

# Inserting a Metadata Function for ASP.NET pages

For a description of the Metadata function, see "MetaData Server Control" on page 185.

This tag appears on the following menu(s):

#### Commands > Ektron CMS400.NET (Server Controls) > 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.

Insert Ektron CMS400 metadata control	
Meta Data.	Insert
Control ID: MetaData2	Cancel
Content ID:	Help
Select the category to retrieve list of content	
Select the content block to insert into the page	
Dynamic Content Parameter: None 💙	

#### 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.
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.

Field	Description
Dynamic Content Parameter	Gets or sets the QueryString parameter to read a content ID dynamically.

```
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.

- 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.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:metadata defaultcontentid="12" id="MetaData2" runat="server"><!--<br-->cms:MetaData&gt;</cms:metadata>
ASP	For ASP pages, see "Inserting a Metadata Function for ASP, CF, JSP or PHP pages" on page 248
ColdFusion	For ColdFusion pages, see "Inserting a Metadata Function for ASP, CF, JSP or PHP pages" on page 248
JSP	For JSP pages, see "Inserting a Metadata Function for ASP, CF, JSP or PHP pages" on page 248
PHP	For PHP pages, see "Inserting a Metadata Function for ASP, CF, JSP or PHP pages" on page 248

### Inserting a Metadata Function for ASP, CF, JSP or PHP pages

For a description of the Metadata function, see "MetaData Server Control" on page 185.

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.

CMS400 Metadata Tag	
Used to display the Metadata information that is assigned to content. Metadata List (with exclusions):	Insert Cancel
Content ID:	Help
Select the category to retrieve list of content	
Select the content block to insert into the page	
Select the Metadata types for exclusion (hold control key for multiple selections)Select the Metadata types for exclusion (hold control key for multiple selections)	
Add to Metadata List	

#### 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.

Field	Description
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.
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>
```

<HEAD>

7. Move that line from the <body> tag into the <head> tag.

<title>WebForm5</title>
<pre><cms:metadata defaultcontentid="12" id="Metadata2" runat="server"></cms:metadata></pre>
<meta content="Microsoft Visual Studio .NET 7.1" name="GENERATOR"/>
<pre><meta content="Visual Basic .NET 7.1" name="CODE_LANGUAGE"/></pre>
<meta content="JavaScript" name="vs_defaultClientScript"/>
<pre><meta content="http://schemas.microsoft.com/intellisense/i&lt;/pre&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;/HEAD&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;body&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;&lt;form id=" form1"="" method="post" name="vs_targetSchema" runat="server"/></pre>
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.



Shown below is an example of the code that is inserted:

Platform	Code View
.NET	For ASP.NET pages, see "Inserting a Metadata Function for ASP.NET pages" on page 245
ASP	<% ecmMetadata "[1;Title][9][3;Keywords,Title]","" %>
ColdFusion	<cfmodule <br="" template="#request.ecm.AppPath#ecmMetadata.cfm">ItemList="[1;Title][9][3;Keywords,Title]"&gt;</cfmodule>
JSP	<%= ecmMetadata("[12;keywords,Title]","") %>
PHP	php ecmMetadata("[1;Title][9][3;Keywords,Title]",""); ?

# **Inserting a Search Function**

For a description of the Search functions, see "Search Server Control" on page 84

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 Function...

To insert a Search 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 **Search**
- 3. The Insert Ektron Search box is displayed.

Insert Ektron CMS400 Search	×
Ektron CMS Search. Control ID:	Search1 Insert Cancel
Folder:	
	Select the folder or fill it in above
Recursive:	
Order By:	Title Ascending 💙
Display Type:	ecmTeaser 🖌
Max Number of results:	0 (0- is unlimited)
ShowSearchBoxAlways:	
Display Horizontal:	
Hide:	
Visible:	
PostResultsAt	<b>#</b>
(Positioning info - GridLayout) Style:	leave blank if none
ButtonImgSrc:	
ButtonText:	Search
EmptyResultMsg:	Search returned zero result
MaxCharacters:	50
TextBoxSize:	0 (If 0 ignore)
Wrap tag:	Span 🖌

#### **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
Control ID	A unique name for the function you are inserting.
Used in ASP.NET only.	
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.
Select Folder	Select a folder path to be searched. If you insert the folder path, you do not need to select a folder.
Recursive	If checked, search information is gathered from the folder selected and all of its child folders.
Order By	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.
	• <b>Title</b> - The title of the content block
	<ul> <li>Date Created - The date the content block was created</li> </ul>
	Date Modified - The date the content block was last modified
	<ul> <li>Last Author Last Name - The last author's last name</li> </ul>
	Last Author First Name - The last author's first name
	Ascending - Alphabetic or numerical order
	Descending - Reverse alphabetic or numerical order
Display Type	Determines how the information on the page is displayed
Used in ASP.NET only.	<b>None</b> -databind only <b>ecmNavigation</b> - lists the title of every content block in the folder
	ecmTeaser - lists the title of every content block in the folder plus the content summary
Max Number of results	Enter a number to specify how many results are displayed on the
Used in ASP.NET only.	

Field	Description
ShowSearchBoxAlways	If <b>unchecked</b> , the search box does not appear on PostBack.
Used in ASP.NET only.	
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.
Hide	Used to hide a search in design time and run time.
Used in ASP.NET only.	Checked = Hide search
	Unchecked = Show search
	Note: If Hide and Visible are both checked, the property is hidden.
Visible	Used to show or hide the function at run time.
Lised in ASP NET only	Checked = Hide function
	Unchecked = Show function
	Note: If Hide and Visible are both checked, the property is hidden.
	The path of the Web page where the energy results will be pagted
Target Page (ASP, CF, JSP or PHP)	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.
	Important: A search control must exist on the posting page.
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page.
Used in ASP.NET only.	For example:
	style="Z-INDEX: 101; LEFT: 328px; POSITION: absolute; TOP: 240px"
ButtonImgSrc	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.
ButtonText	The button text is the text that is displayed on the standard submit button, or the alt text of a search button image.

Field	Description	
EmptyResultMsg	The return message displayed, if the search returns no hits.	
Used in ASP.NET only.	Search returned zero results	
MaxCharacters	Specify the maximum amount of characters that is accepted in the text box.	
TextBoxSize	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.	
Font Face	The type of font to use for search function.	
Used in ASP, CF, JSP and PHP only.		
Font Color	The color of the font used in the search function.	
Used in ASP, CF, JSP and PHP only.		
Font Size	The size of the font used in the search function.	
Used in ASP, CF, JSP and PHP only.		
Wrap Tag	Allows a developer to specify a server control's tag.	
Used in ASP.NET only.	Span - The <span. Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></span. 	
	<b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.</div>	

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	<cms:search <br="" displayxslt="ecmTeaser" id="Search1" recursive="False">FolderPath="\" runat="server"&gt;</cms:search>
ASP	<% ecmSearch "\Products",1,"SearchDisplay.asp",10,25,"","Search", "Verdana","Blue","2",0,"" %>
ColdFusion	<cfmodule <br="" 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;</cfmodule>
JSP	<%= ecmSearch("\Products",1,"SearchDisplay.jsp",10,25,"","Search", "Verdana","Blue","2",0,"") %>
PHP	php ecmSearch("\Products",1,"SearchDisplay.php",10,25,"","Search",<br "Verdana","Blue","",0,""); ?>

# **Inserting a Search Display Function**

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 Displ	ay Tag	
Creates area where search results Maximum Number of Documents:	s will be displayed.	Insert Cancel
Style Info:		Help
Show Last Modified Date:		
Insert Default Content:		
Default Content ID:		
Select the	category to retrieve list of content	
Select the	content block to insert into the page	

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.

Field	Description
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.
Insert Default Content	Specify the ID number of the content block that you wish to use as a default content block on the Web page when no search is executed.
Default Content ID	If you are working online, you may select the content block to use as a default content block from the list.

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	<% ecmSearchDisplay 0, "", 1 %>
ColdFusion	<cfmodule <br="" template="#request.ecm.AppPath#ecmSearchDisplay.cfm">MaxNumber="0" StyleInfo="" ShowDate="1"&gt;</cfmodule>
JSP	<%= ecmSearchDisplay(0, "", 1)%>
PHP	php ecmSearchDisplay(0, "", 1)?

# **Inserting a Single Summary Function**

For a description of the single summary function, see "Single Summary Function" on page 364

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 **Insert Single Summary Tag...**

Ektron CMS400 Single Summary Tag		
Used to display a content block summary. Content ID:Select the category to retrieve list of content	Insert Cancel Help	

### 3. The Insert Ektron Single Summary Tag box is displayed.

4. Use the following table to assist you with inserting a single summary tag.

Field	Description
Content ID	<ul> <li>Working Offline - Enter a valid content ID into the Content ID textbox.</li> <li>Working Opline - The ten drep down list will be filled with a list of content fold</li> </ul>
	• Working Online - The top drop-down list will be filled with a list of content folderers 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.
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.

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 <br="" template="#request.ecm.AppPath#ecmSingleSummary.cfm">ContentID="1" ShowSummary="1" StyleInfo="" ShowInfo="DateModified"&gt;</cfmodule>
JSP	<%= ecmSingleSummary("1",1,"","DateModified","") %>
PHP	php ecmSingleSummary("1",1,"","DateModified","") ?

### **Inserting a Random Content Function**

For a description of the random content function, see "Random Content" on page 394

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.

Ektron CMS400 Random content Tag	
Random item of a collection.         Collection ID:	Insert Cancel Help
XSLT to Apply: Select if an XLST is to be applied 💙	

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 XML Content only	If desired, you may specify an XSLT to apply to the content that gets displayed.
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
ASP	<% ShowRandomContent 3, 0 %>
ColdFusion	<cfmodule <br="" displayfunction="0" id="3">template="#request.ecm.AppPath#ecmShowRandomContent.cfm"&gt;</cfmodule>
JSP	<%= ShowRandomContent(3, 0) %>
PHP	php ShowRandomContent(3, 0); ?

## **Inserting a Random Summary Function**

For a description of the random summary function, see "Random Summary" on page 395

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 **Insert Random Summary Tag...**
- 3. The Insert Ektron CMS400.NET Random Summary Tag box is displayed.

Ektron CMS400 Random summary Tag	
Random summary of an item in a collection. Collection ID:	Insert Cancel
Select the collection to insert into the page	

# 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 summary function will retrieve summaries from.

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
ASP	<% ShowRandomSummary (2) %>
ColdFusion	<cfmodule id="2" template="#request.ecm.AppPath#ecmShowRandomSummary.cfm"></cfmodule>
JSP	<%= ShowRandomSummary(2) %>
PHP	php ShowRandomSummary (2); ?

### **Inserting an Event Calendar Function**

For a description of the Event Calendar function, see "Calendar Server Control" on page 126.

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 Function...

To insert an event 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.

Insert Ektron cms400 Calenda	r	
Displays and manages a content blo	ck.	Insert
Control ID:	Calendar 1	Cancel
Content ID:		Help
Select the calendar to in	sert into the page 💙	
Dynamic Content Parameter:	None 💌	
CalDisplayType:	Month 🗸	
ShowXML:		
Hide:		
Visible:		
(Positioning info - GridLayout) Style:	leave blank if none	
Wrap tag:	Span 🗸	

# **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.

Ektron CMS400 Calendar Tag	
List of Calendars. Calendar ID:	Insert Cancel
Select the calendar to insert into the page	

#### 4. Use the following table to assist you with inserting a calendar.

Field	Description
Control ID	A unique name for the function you are inserting.
Used in ASP.NET only.	
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	To make this calendar dynamic, select <b>calendar_id</b> . When you do, this server control uses the calendar passed as a URL parameter.
Used in ASP.NET only.	
CalDisplayType	Specify the Calendar view type. The default is <b>Month</b> .
Used in ASP.NET only.	• Month - A 30 day calendar month view.
	• Day - Single day view.
	• Inline - Display events that occur from the start date of the calendar through the end date.
	• <b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.

Field	Description
ShowXML Used in ASP.NET only.	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>
	Checked = Show raw XML data
	Unchecked = Show Calendar
Hide	Used to hide a calendar in design time and run time.
Used in ASP.NET only.	Checked = Hide calendar
	Unchecked = Show calendar
	Note: If Hide and Visible are both checked, the property is hidden.
Visible	Used to show or hide the function at run time.
Used in ASP.NET only.	Checked = Hide function
	Unchecked = Show function
	Note: If Hide and Visible are both checked, the property is hidden.
Style (Positioning Info - GridLayout)	Set position information of a control as absolute, when using a GridLayout Web page.
Used in ASP NET only	For example:
	style="Z-INDEX: 101; LEF1: 328px; POSITION: absolute; TOP: 240px"
Wrap Tag	Allows a developer to specify a server control's tag.
Used in ASP.NET only.	The default is <b>Span</b> . <b>Span</b> - The <span> tag is used to designate an in-line portion of an HTMI</span>
	document as a span element.
	<b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.</div>

A graphic defines where the calendar appears on your template.

Calend	darAPI.aspx*
	Code 🔄 Split 📴 Design 🥳 Title: WebForm2 🖓 🦺 🌍 C 📰
22 <	(body>
23	<form id="Form1" method="post" runat="server"></form>
24	<p> </p>
25	<pre><cms:calendar defaultcalendarid="1" id="Calendar1" runat="server"></cms:calendar></pre>
26	
<	
ekt	tron Calendar

Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<cms:calendar defaultcalendarid="1" id="Calendar1" runat="server"><!--<br-->cms:Calendar&gt;</cms:calendar>
ASP	<% ecmEvtCalendar (1) %>
ColdFusion	<cfmodule <br="" calendar_id="1">template="#request.ecm.AppPath#ecmEvtCalendar.cfm"&gt;</cfmodule>
JSP	<%= ecmEvtCalendar(1) %>
PHP	php ecmEvtCalendar (1); ?

### **Inserting a Form**

For a description of the Form function, see "FormBlock Server Control" on page 80

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 Function...

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.

Insert Ektron CMS400 Form		
Displays and manages a content block. Control ID	FormBlock1	Insert Cancel
Form ID::		Help
Select the Form to insert into	he page 💉	
Dynamic Form Parameter	None 🗸	
Add Validation: (default true		
Include Form Tags: (default false		
Hide	:	
Visible	. 🗹	
(Positioning info - GridLayout) Style	leave blank if none	
Wrap tag	: Span 💌	

 _		_			_
 <b>D</b> /	<u> </u>	דר	• •		т
~		< I	-	<b>N</b>	
 	•••	•••	<u> </u>		

When inserting a form tag using ASP, CF, JSP or PHP, a different dialog box appears.

List of Forms.	Insert
Form ID:	Cancel
Select the Form to insert into the page	Help

Field	Description	
Control ID	A unique name for the function you are inserting.	
Used in ASP.NET only.		
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	To make this form block dynamic, select <b>ekfrm</b> . When you do, this control uses the form block passed as a URL parameter	
Used in ASP.NET only.		
Add Validation	When <b>checked</b> , adds Validation to your main runat=server form.	
Used in ASP.NET only.		
Include Form Tags	Determines if tags are generated automatically or manually. See Also: "Automatic versus Manual Generation of Form Tags" on page 83	
Used in ASP.NET only.		
Hide	Used to hide a form in design time and run time.	
Used in ASP.NET only.	Checked = Hide form	
	Unchecked = Show form	
	Note: If Hide and Visible are both checked, the property is hidden.	
Visible	Used to show or hide the function at run time.	
Used in ASP.NET only.	Checked = Hide function	
	Unchecked = Show function	
	Note: If Hide and Visible are both checked, the property is hidden.	

4.	Use the following	table to	assist you	with in	nserting a	form.
----	-------------------	----------	------------	---------	------------	-------

Field	Description
Style (Positioning Info - GridLayout) 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 Used in ASP.NET only.	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> </ul>

A graphic defines where the form appears on your template.

FormBlock.aspx*
Code 🖴 Split 📴 Design 📝 Title: WebForm1 🖓 🖧 🌍 C 📃
15 < <u>P&gt; </u>
16 <pre>(<cms:formblock defaultformid="13" id="FormBlock1" runat="server"></cms:formblock>)</pre>
17
18 <p> </p>
ektron Form Block
Description

#### Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<cms:formblock defaultformid="13" id="FormBlock1" runat="server"><!--<br-->cms:FormBlock&gt;</cms:formblock>
ASP	<% ecmFormBlock (1) %>
ColdFusion	<cfmodule id="1" template="#request.ecm.AppPath#ecmFormBlock.cfm"></cfmodule>

Platform	Code View
JSP	<%= ecmFormBlock(1) %>
PHP	php ecmFormBlock (1); ?

### Inserting a 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.

CMS400 Multipurpose Tag	
Displays a content block or a form block. Select a default content id to display if the content id or form id is not passed from URL parameter.	Insert Cancel
Select the category to retrieve list of content	
Select the content block to insert into the page 💙	

# 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.

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	<% ecmMultiPurpose(24) %>
ColdFusion	<cfmodule <br="" defaultid="24">template="#request.ecm.AppPath#ecmMultipurpose.cfm"&gt;</cfmodule>
JSP	This function is not available as an Ektron Dreamweaver Extension for JSP.
PHP	php ecmMultiPurpose(12); ?

## Inserting an Advanced Calendar Tag

This tag appears on the following menu(s):

 Commands > Ektron CMS400 - (ASP CF JSP PHP) > CMS Function...

To insert an Advanced Calendar 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 Advanced Calendar...

3. The Insert Ektron CMS400.NET Advanced Calendar Tag box is displayed.

Ektron CMS400 Advance Calendar	
List of Calendars. Calender ID:	Insert Cancel
Select the calendar to insert into the page 😪	Help
Show Events Type:	

4. Use the following table to assist you with inserting a Advanced Calendar tag.

Field	Description
Calendar ID	Type in the calendar ID, or select a calendar from the drop down list
Show Event Types	Shows a list of event types that can be used to sort events on the calendar. The default is checked.
	Checked - the list of event types appears with the calendar.
	Unchecked - the list of event types does not appear with the calendar.

5. When completed, click insert.

Shown below is the code that will be inserted according to the platform that has been specified in the setup:

Platform	Code View
ASP	<% ecmAdvancedEvtCalendar 1, 1 %>
ColdFusion	<cfmodule <br="" calendar_id="3" showevents="1">template="#request.ecm.AppPath#ecmAdvanceEvtCalendar.cfm"&gt;</cfmodule>
JSP	<%= ecmAdvancedEvtCalendar (1, 1) %>
PHP	php ecmAdvancedEvtCalendar(1, 1); ?

### Inserting an XML Index Search

For a description of the XML Index Search Function, see "IndexSearch Server Control" on page 96.

This tag appears on the following menu(s):

 Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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.

Ektron CMS400 Index Search	
Used to display a list of content block from a folder.	Insert
Control ID:	Cancel
XML ConfigID:	Help
Select the XML Configuration	
FolderID:	
Select the folder or fill it in above	
Search Parm XML:	
Weighted: 🔽	
ShowSearchBoxAlways: 🔽	
Recursive: 🔽	
Order By: Title 🗸 Ascending 🗸	
Max Number of results: 0	
Display Type: ecmTeaser 💉	
Dynamic Content Parameter: None 💉	
Hide:	
Visible: 🔽	
(Positioning info - GridLayout) Style: leave blank if none	
Wrap tag: Span 🔽	
Button Text: Search	
Hits Marker: *	

3. The Insert XML Index Search box is displayed.

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 XML configuration whose indexed fields appear on the search screen. If you do not know the XML ConfigID, you can navigate to it using the Select XML Configuration 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	If this value is <b>checked</b> , content blocks are ranked by the number of criteria matches.
	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.
	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.
	Checked- display every content block that matches at least one search criteria
	Unchecked - show only content blocks that match all search criteria
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.

Field	Description
Order By	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.
	• Title - The title of the content block
	Date Created - The date the content block was created
	Date Modified - The date the content block was last modified
	Last Author Last Name - The last author's last name
	Last Author First Name - The last author's first name
	Ascending - Alphabetic or numerical order
	Descending - Reverse alphabetic or numerical order
	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.
Max Number of results	The Maximum number of content blocks returned (0=unlimited).
Display Type	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 <i>See Also:</i> "Example of ecmNavigation Display" on page 121 <b>ecmTeaser</b> - lists the title of every content block in the folder plus the content
	summary See Also: "Example of ecmTeaser Display" on page 122
Dynamic Content Parameter	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. <b>None</b> - Use Default <b>ID</b> - The control uses the content block passed as a URL parameter.
Hide	Used to hide a form in design time and run time.
	Checked = Hide form
	Unchecked = Show form
	Note: If Hide and Visible are both checked, the property is hidden.

Field	Description
Visible	Used to show or hide the function at run time.
	Checked = Hide function
	Unchecked = Show function
	Note: If Hide and Visible are both checked, the property is hidden.
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.</div></span>
Button Text	The text that appears on the button.
Hits Marker	This symbol shows how many search criteria each hit matched. The following example shows the first hit met three of the criteria. The second group met one criterion.

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	<cms:indexsearch <br="" id="IndexSearch1" xmlconfigid="4">DisplayXslt="ecmTeaser" HitsMarker="*" runat="server"&gt;<!--<br-->cms:IndexSearch&gt;</cms:indexsearch>
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.

## **Inserting Display By Meta Value**

For a description of the Display By Meta Value function, see "MetaDataList Server Control" on page 188.

This tag appears on the following menu(s):

• Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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.

Ektron CMS400 Display By Met	a Value	
Used to display a list of content block	by Meta Valuer.	Insert
Control ID:	Metadatalist1	Cancel
FolderID:		Help
§	Select the folder or fill it in above 💌	
Recursive:		
Order By:	Title 🖌 Ascending 🗸	
Max Number of results:	0	
ExactPhrase:		
KeyWordName:		
KeyWordValue:		
Display Type:	ecmTeaser 👻	
Dynamic Content Parameter:	None 💌	
GetHtml:		
Content Type:	Content 👻	
Hide:		
Visible:		
(Positioning info - GridLayout) Style:	leave blank if none	
Wrap tag:	Span 🖌	

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.
Recursive	Determines whether the Display By Meta Value control uses this folder's child folders.
Order By	• Specify how you want to order the results that are generated. Also specify if you want the results in ascending or descending order.
	• <b>Title</b> - The title of the content block
	Date Created - The date the content block was created
	Date Modified - The date the content block was last modified
	Last Author Last Name - The last author's last name
	Last Author First Name - The last author's first name
	ID - The content blocks ID
	Ascending - Alphabetic or numerical order
	Descending - Reverse alphabetic or numerical order
Max Number of results	The Maximum number of content blocks returned (0=unlimited).
ExactPhrase	Determines whether the KeyWordValue needs to match the phrase exactly. For example, if "site" is the KeyWordValue and the title of a content block is "Welcome to the site" and ExactPhrase is set to true, you would not see the content block in the metadata list. This is because "site" does not equal "Welcome to the site". Checked = Match the exact phrase
KeyWordName	The KeyWordName is the container where the KeyWordValues are located.
	Some examples of a KeywordName might be "Keywords or Title."
	CMS400.NET Administrators Manual > "Managing Content" > "Metadata" > "Managing Metadata Definitions" > "Adding a Metadata Definitions"

Field	Description
KeyWordValue	The value that is associated with the KeyWordName.
	Some examples of a KeyWordValue might be "Home; page; or company"
Display Type	Determines how the information on the page is displayed.
	None-databind only
	ecmNavigation - lists the title of every content block in the folder
	See Also: "Example of ecmNavigation Display" on page 121
	ecmTeaser - lists the title of every content block in the folder plus the content summary
	See Also: "Example of ecmTeaser Display" on page 122
Dynamic Content Parameter	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. <b>None</b> - Use Default
	ID - The control uses the content block passed as a URL parameter.
GetHtml	Set to <b>True</b> 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 DataGrid.
Content Type	Defines what type of list summary appears.
	<b>Content</b> - Displays only content blocks in the folder.
	All Types - Displays all content in the folder.
	Forms - Displays only forms in the folder.
	Archive_Content - Displays archived content from the folder.
	Archive_Forms - Displays archived forms from the folder.
Hide	Used to hide the function in design time and run time.
	Checked = Hide function
	Unchecked = Show function
	Note: If Hide and Visible are both checked, the property is hidden.

Field	Description
Visible	Used to show or hide the function at run time.
	Checked = Hide function
	Unchecked = Show function
	Note: If Hide and Visible are both checked, the property is hidden.
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.</span>
	<b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.</div>

A graphic defines where the Display By Meta Value appears on your template.

DisplayCo	ontentByMeta.aspx*	
Cor	de 📇 Split 🔄 Design 📝 Title: DisplayContentByMeta 🖓 🕀 🊱 🥂	
13	<body></body>	
14	<form id="Form1" method="post" runat="server"></form>	
15	<p> </p>	
16	<cms:metadatalist <="" folderid="6" id="Metadatalist1" td=""></cms:metadatalist>	
17	DisplayXslt="ecmTeaser" runat="server">	
18		
<		
ektron display by Metadata Description		

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:metadatalist <br="" folderid="69" id="Metadatalistl">DisplayXslt="ecmTeaser" runat="server"&gt;</cms:metadatalist>
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.

## **Inserting a Standard Menu**

For a description of the Standard Menu function, see "Menu Server Control" on page 160.

This tag appears on the following menu(s):

 Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

To insert a Standard 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 **Standard Menu**

The Standard Menu Cor	ntrol box appears.
-----------------------	--------------------

Ektron CMS400 Menu Control	
Menu Control ID: Menu1	Insert Cancel
DefaultMenuID:	Help
Select the from a list of menus	
MenuType: SampleMenu 🗸	
Dynamic menu Parameter: None 🗸	
Hide:	
Visible: 🔽	
(Positioning info - GridLayout) Style:	
Wrap tag: Span 🖌	

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.
DefaultMenulD	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.
МепиТуре	The type of menu to render. <b>None</b> - databind only <b>SampleMenu</b> - A sample display, formatted as a bulleted menu list <b>TreeMenu</b> - A sample display, formatted as a folder tree. You can expand the tree by clicking on the folder icon. For more information on using the SampleMenu and TreeMenu Display, see "Using DisplayXslt Samples" on page 163.

Field	Description
Dynamic Menu Parameter	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.
Hide	Used to hide a menu in design time and run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
Visible	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
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.</div></span>

A graphic defines where the Standard Menu appears on your template.

menu.aspx*		
🛛 🐼 Code 📇 Split 🔄 Design 🥳 Title: menu 🖓 🕀 🌚 (		
14 <p> </p>		
15 <pre>(<cms:menu defaultmenuid="15" id="Menu1" runat="server"></cms:menu>)</pre>		
16		
ektron Menu		
Description		

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:menu defaultmenuid="15" id="Menul" runat="server"></cms:menu>
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.

## Inserting a DHTML Menu

For a description of the DHTML Menu function, see "DhtmlMenu Server Control" on page 177.

This tag appears on the following menu(s):

 Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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:	DhtmlMenu1 Cancel
DefaultMenuID:	Help
Select the from a list of r	nenus 🖌
Align:	Center 💌
CssClass:	
HorizontalDisplay:	
MenuHoverOverColor:	#6699CC
SubMenuBackColor:	#CCCCDD
SubMenuHoverOverColor:	#6699CC
BackColor:	#92BED7
ForeColor:	
Font-Bold:	
Font_Italic:	
FontNames:	
Font-Size:	
SubMenuItemHeight:	20
SubMenuItemWidth:	150
SubMenuTopOffset:	3
MenuStyle:	
Stylesheet:	
Dynamic menu Parameter:	None

# 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.	
DefaultMenulD	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	<ul> <li>Specifies the horizontal alignment of the root menu text.</li> <li>The default is Center.</li> <li>NotSet - Use the default setting.</li> <li>Left - Aligns text to the left.</li> <li>Center - Aligns text to the center.</li> <li>Right - Aligns text to the right.</li> <li>Justify - Aligns text justified to both margins.</li> </ul>	
CssClass	Apply a .css class to the root menu. Enter the class name from the .css defined in the Style sheet property.         Since pop_style.css is the default style sheet, you can use its classes without defining it in the Stylesheet property.         For example, if you enter lowtext in the property, the root menu changes as shown below.         No CssClass applied       lowT ext CssClass applied         MenuExample       MenuExample	

Field	Description
HorizontalDisplay	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. The default is False True = Submenu appears to the right of the root menu $\underbrace{Our \ Company} \ Company Directory} \\ \hline \\ \hline \\ Root \ Menu \ \hline \\ Veb \ Forms \\ Latest \ Changes \\ Sitemap \ \hline \\ Submenu \ \hline \\ \\ Submenu \ \hline \\ \hline \\ Submenu \ \hline \\ \\ \hline \\ Submenu \ \hline \\ \\ \hline \\ Submenu \ \hline \\ \\ \\ \hline \\ \hline \\ \\ \hline \\ \hline \\ \\ \hline \hline \\ \hline \\ \hline \hline \\ \hline \\ \hline \hline \hline \\ \hline \hline \\ \hline \hline \hline \\ \hline \hline \hline \hline \\ \hline \hline \hline \\ \hline \hline$
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.

Field	Description		
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> .		
FontNames	Enter the name of a font to use.		
Font-Size	The size of the font.		
SubMenuItemHeight	Sets the height of each submenu item in pixels. The Default is 20		
	Example set to 20: CMS Developer		
	Example set to 30: CMS Developer		
SubMenuItemWidth	Sets the width of each submenu item in pixels. The Default is 150		
	Example set to 150: CMS Developer		
	Example set to 130: CMS Developer		
SubMenuTopOffset	Sets the location of the top of the submenu. This number is in pixels. The default is 5		
	Example set to 5:		
	Book > ASP.NET Unleashed		
	News > Programming C#.NET		
	VB.NET How to		
	Example set to 0 (zero):		
	Book > ASP.NET Unleashed		
	News > Programming C#.NET		
	VB.NET How to		
	Example set to -10:		
	Book ASP.NET Unleashed		
	News > Programming C#.NET		
	VB.NET How to		

Field	Description
MenuStyle	A specified inline style that defines the root menu. For example, if you enter the following in the MenuStyle property box: font: bold 18px Arial;color:royalblue; background- color:slateblue
	our company
StyleSheet	Specify the location and style sheet for the submenu. For example: \Workarea\csslib\mytest.css.
	Leave blank to use the default.
	pop_style.css is the default style sheet. It is located in \webroot\CMS400Example\Workarea\csslib. For more information on style sheets, see "Using a Style Sheet" on page 182
	The Stylesheet property can be used in conjunction with the CssClass property to define the root menu.
Dynamic Menu Parameter	Checks the QueryString for the menu value and replaces the menu with content when ID is specified. Leave blank to always display the menu.
	None - Use Default
	menu_ID - The control uses the menu passed as a URL parameter.

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	<cms:dhtmlmenu <br="" defaultmenuid="16" forecolor="" id="DhtmlMenu1">runat="server"&gt;</cms:dhtmlmenu>
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.

## Inserting a Language Select Box

For a description of the Language Select Box, see "LanguageSelect Server Control" on page 65.

This tag appears on the following menu(s):

• Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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.

Insert Ektron CMS400 Languag	e Select	
Language Select Control. Control ID:	LanguageSelect1	Insert Cancel
Hide: Visible:		Help
(Positioning info - GridLayout) Style:	leave blank if none	
Wrap tag:	Span 🔽	

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.

Field	Description
Hide	Used to hide a function in design time and run time.  Checked = Hide function  Unchecked = Show function  Note: If Hide and Visible are both checked, the property is hidden.
Visible	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.
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	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> </ul>

A graphic defines where the Language Select appears on your template.

LanguageAPI.aspx*		
🕼 🖸 Code 📇 Split 📑 Design 📝 Title: LanguageAPI	5 <mark>4</mark> 0	î. (5). C 🗉.
15 <p> </p>		
<pre>16 &lt;<cms:languageselect id="LanguageSelect1&lt;/pre&gt;&lt;/th&gt;&lt;th&gt;" runat="server">&gt;</cms:languageselect></pre>	/cms:LanguageSelect>	
17		
< <u>.</u>		
Language: English (U.S.) v		
	Default Language:	[A SD-L iteral]

Shown below is an example of the code that is inserted.

Platform	Code View
.NET	<cms:languageselect id="LanguageSelect1" runat="server"><!--<br-->cms:LanguageSelect&gt;</cms:languageselect>
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.

## Inserting a Language API

For a description of the Language API function, see "LanguageAPI Server Control" on page 67.

This tag appears on the following menu(s):

• Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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.

Insert Ektron CMS400 Languag	ge API	
Language API Control.		Insert
Control ID:	LanguageAPI2	Cancel
Hide:		Help
Visible:		
(Positioning info - GridLayout) Style:		
	leave blank if none	
Wrap tag:	Span 💙	

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	Used to hide a Language API in design time and run time.  Checked = Hide Language API Unchecked = Show Language API Note: If Hide and Visible are both checked, the property is hidden.
Visible	Used to show or hide the function at run time.  Checked = Hide function Unchecked = Show function Note: If Hide and Visible are both checked, the property is hidden.

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	<ul> <li>Allows a developer to specify a server control's tag.</li> <li>The default is Span.</li> <li>Span - The <span> tag is used to designate an in-line portion of an HTML document as a span element.</span></li> <li>Div - The <div> tag is used when you want to apply attributes to a block of code.</div></li> </ul>

5. When completed, click **insert**.

A graphic defines where the Language API appears on your template.

LanguageAPI.aspx*	
Code 🔛 Split 🔄 Design 📝 Title: LanguageAPI 🖓 🕅	6 C
15 <p> </p>	
16 <cms:languageapi id="LanguageAPI2" runat="server"><th>guageAPI&gt;</th></cms:languageapi>	guageAPI>
17	
ektron Lang-API	
Default Language:	[ASD-Literal

Default I annuage [ASD-Literal]

Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:languageapi id="LanguageAPI2" runat="server"></cms:languageapi>
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.

### Inserting an RSS Aggregator

For a description of the RSS Aggregator function, see "RssAggregator Server Control" on page 193.

This tag appears on the following menu(s):

• Commands > Ektron CMS400.NET (Server Controls) > CMS Function...

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.

Ektron CMS400 RSS Aggregator	r	
Used to display an RSS feed.		Insert
Control ID:	RssAggregator 1	Cancel
URL:		Help
Max Number of results:	0	
Display Type:	ecmTeaser 🖌	
Hide:		
Visible:		
(Positioning info - GridLayout) Style:		
Wrap tag:	leave blank if none Span 👻	

# 4. Use the following table to assist you with inserting an RSS Aggregator.

A unique name for the function you are inserting. The RSS feed path for the server control. For example: http://msdn.microsoft.com/rss.xml
The RSS feed path for the server control. For example: http://msdn.microsoft.com/rss.xml
The Maximum number of items from an RSS feed that are returned (0=unlimited).
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 Also: "Example of ecmNavigation Display" on page 121
ecmTeaser - lists a title and a description of every RSS feed item
See Also: "Example of ecmTeaser Display" on page 122
Path to Custom Xslt - Enter the path to an Xslt that determines the display of the page
Used to hide a Language API in design time and run time.
Checked = Hide
Unchecked = Show Language API
Note: If Hide and Visible are both checked, the property is hidden.
Used to show or hide the function at run time.
Checked = Hide function
Unchecked = Show function
Note: If Hide and Visible are both checked, the property is hidden.
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	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.</span>
	<b>Div</b> - The <div> tag is used when you want to apply attributes to a block of code.</div>

5. When completed, click insert.

A graphic defines where the RSS Aggregator appears on your template.



#### Shown below is an example of the code that is inserted:

Platform	Code View
.NET	<cms:rssaggregator displayxslt="ecmTeaser" id="RssAggregatorl" runat="server" url="http://msdn.microsoft.com/&lt;br&gt;rss.xml"></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.

## XML Functions...

In addition to the many CMS functions, an additional XML related function may be inserted into your Web site templates. The following section explains this function.

#### **Accessing XML Function Commands**

To access the CMS XML Functions command menu, follow this path:

Commands > Ektron CMS400 - (ASP CF JSP PHP) > XML Functions...

A list of all CMS XML Functions that can be inserted is displayed.

	5 91 3		
y <u>T</u> ext	<u>Commands</u> <u>Site</u> <u>Window</u> <u>Help</u>		
out   For	Start <u>R</u> ecording Play <u>R</u> ecorded Command <u>E</u> dit Command List <u>G</u> et More Commands <u>M</u> anage Extensions	Ctrl+Shift+X	Favorites
sign (380) k(380) he line "admin"	Apply Source Formatting Apply Source Formatting to Selection Clean Up HTML Clean Up Word HTML Add/Remove Netscape Resize Fix		C E.
	Optimize Image in Fireworks Greate Web Photo Album Se <u>t</u> Color Scheme Eormat Table		
	Sort Table		
	Ektron CMS400-(ASP CF JSP PHP) Ektron CMS400.NET (Server Controls)	• •	Setup         CMS Function         XML Function         Content XSLT Tag
			λ <del>ζ</del>

The following section explains how to insert the XML functions.

### Insert 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.

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.

Insert Ektron CMS300 Content and apply an XSLT	X
Displays and manages a content block with an XSLT. Content ID:	Insert Cancel
Select the category to retrieve list of contentSelect the category to retrieve list of content	Help
Select the content block to insert into the page	
Select how to pick the XSLT  The XSLT must have a fully qualified path like: C:\somepath\myxslt.xsl XSLT:	

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.
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
ASP	<% ecmContentBlockEx 5, "", 1 %>
ColdFusion	<cfmodule <br="" id="5" xsltpath="" xslttype="1">template="#request.ecm.AppPath#ecmContentBlockEx.cfm"&gt;</cfmodule>
JSP	<%= ecmContentBlockEx(24, "", 0 ) %>
PHP	php ecmContentBlockEx (5, "", 1); ?

# 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, databindable ASP.NET controls, Ektron empowers developers working in Visual Studio .NET 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 24.

The following examples shows the Login Server Control added by drag and drop vs. adding a Login method by the ASP.NET method.

Toolbox <b>4</b>	×	WebForm2.aspx*	WebForm2.aspx.vb*
Data			
Web Forms		🛛 🖉 Login 🗋 🗆	
Components			
HTML			
Clipboard Ring			
General			
ektron			
🕨 Pointer			
🧾 Calendar			
🛱 Collection			
📓 ContentBlock			
⁄ FormBlock			
🎲 LanguageSelect			
😑 ListSummary			
💦 Login 🔪			
📜 Menu 🛛 🗟			
🔎 Search			

Drag and Drop Login Server Control

Adding a Login function via an ASP.NET method.

<td< td=""><td>&gt;</td></td<>	>
	<%=AppUI.ecmLogin()%>
<td>d&gt;</td>	d>

#### **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.

**NOTE** 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 Inetpub/wwwroot/CMS400Example/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 .NET.
- 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 Ektron.CMS.UI.CommonUI to display information from the .net assembly file
<% Dim AppUI As New ApplicationAPI %>	Creates an instance of the ApplicationUI class that can be 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	"Add Content" on page 312
Collection	Displays links generated by collections.	"Collection Server Control" on page 119
Content Block	Displays and manages a content block.	"ContentBlock Server Control" on page 70
Event Calendar	Displays an event calendar	"Calendar Server Control" on page 126
Form	Displays a content block that is associated with a form.	"FormBlock Server Control" on page 80
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 96

Method	Description	More Information
List Summary	Displays a list of content block summaries.	"ListSummary Server Control" on page 113
Login	Displays the login or logout button, which lets the user log in and use Ektron CMS400.NET.	"Login Server Control" on page 63
Metadata	Displays the metadata in the source of the Web page displaying the content.	"MetaData Server Control" on page 185
Menu	Customizes the DHTML menus feature	"DhtmlMenu Server Control" on page 177
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 84
ShowSiteLanguage	Lets a site visitor select a language in which to view site.	"LanguageSelect Server Control" on page 65

# Add Content

The add content 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

Here is the format for the add content method.

<% AddContent ( UserName, Password, Domain, content\_title, content\_comment, ContentHtml, SummaryHTML, MetaInfo, ContentLanguage, FolderID, GoLive, EndDate, ErrString )%>

This function is basically the same as the Add Content function in ASP. See "Add Content" on page 329 for details.

The only difference between the ASPX function and the ASP one is the syntax for implementing it.

### Add Content Example

The following is an example of the add content method hard coded in a template.

```
<</td>
```

# **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/CMS400Example/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.myektronsite.com, the WSDLpath would be http://www.myektronsite.com/WorkArea/WebServices/ContentWS.asmx.

To customize Ektron CMS400.NET, you need to be familiar with these tags.

Function	Description More Information	
Calendar Functions		
Calendar	Displays a calendar with different display types. For example, a daily view calendar.	"Calendar Function" on page 399
Event Calendar	Displays a month view event calendar "Event Calendar Function" on pa	
Collection Functions	3	
Collection	Displays links generated by collections	"Collection Function" on page 374

Function	Description	More Information	
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 389	
Random Content	Randomly displays the content of a collection item	"Random Content" on page 394	
Random Summary	Randomly displays the summary of a collection item	"Random Summary" on page 395	
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 387	
Content Functions			
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> progammatically.	"Add Content" on page 329	
Content Block	Displays and manages a content block	"Content Block Function" on page 319	
Form	Displays a content block associated with a form	"Form Function" on page 403	
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 406	
List Summary Funct	ions		
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 363	
List Summary	Displays a list of content block summaries	"List Summary Function" on page 350	
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 358	
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS Summary Function" on page 368	

Function	Description	More Information
Single Summary	Displays the summary for a content block	"Single Summary Function" on page 364
Login Functions		
Login	Adds the login or logout button	"Login Function" on page 317
Menu Functions		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 416
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 416
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu" on page 421
Metadata Functions		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 407
Search Functions		
Search	Inserts a text box with a search button	"Search Functions" on page 333
Search Display	Creates an area to display search results	"Search Display Function" on page 344
Site Language Funct	tions	
ShowSiteLanguage	Lets a site visitor select a language in which to view site.	"Show Site Language Function" on page 415

# **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	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	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	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
✓Preview	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.
× Preview	Preview Off	Turns off preview mode.

## Login Tag Example

The following example places a login button in a table cell inside an asp template.

<% ecmLogin %>

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 an XML Configuration" > "Adding an XML Configuration 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	%>

Νοτε

**E** 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 406.

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: "The Function's Arguments" on page 326	No
Override	If no XSLT is specified, then enter 1 to use XSLT1, 2 for XSLT2, or 3 for XSLT3. See Also: "The Function's Arguments" on page 326	No

In normal view, these tags retrieve the content block from the database and displays it in the template.



#### Welcome to RC International!

RC International is dedicated to the RC racing enthusiasts! We eat, work, play, and live R years RC International has become one of the leading manufactures of RC racing and flyin to the sport, and the enthusiasts who play it, has endeared our products to the RC comm striving to improve our affordable products and hope you will become an RC International

#### NEW FEATURE:Trinity In-Line Pipe

#### NEW PRODUCT: Introducing th

Trinitys new pipe and manifold are for rear exhaust engines and are now used in both the Sportster and the cheetah. Both the manifold and pipe are made from high-quality aluminum. This new pipe is more durable and will withstand a bad crash.

RC International is proud to intr This new addition to the RC Int new options to our customers. wings and a wing span of 71", t has to be seen. 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.



## **Content Block Examples**

Ektron CMS400.NET has two types of content blocks.

Туре	Displays	More information
Static	One content block on the Web page	"Static Content Block" on page 323
Dynamic	A content block whose ID is passed through the URL	"Dynamic Content Block" on page 324

In addition, a content block can contain XHTML or XML content. For more information, see "XML Content Block" on page 325.

#### **Static Content Block**

The following illustrates how to insert a static content block on a template.

<% ecmContentBlock(329) %>

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.

<% ecmContentBlock(request.QueryString("id")) %>	

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.

é	Emp	oyme	ent Oppo	ortu	nities -	Micro	osoft II	nter	net I	Ехріо	rer		į.				
	Eile	<u>E</u> dit	⊻iew	F <u>a</u> v	/orites	Tool	s <u>H</u> elp	2									
	G	Back	-0	~	*				Searc	2	31	Favorite	es 🄇	🕑 Me	dia	Ø	
1	A <u>d</u> dr	ss 🙋	http://1	27.0	.0.1/cms	samp	le/index	.asp	?id=1	.4							
	R	C	111	ie	<i>.</i>	a	lio	1	al	ſ							
Ľ	lome	P	roducts		Suppo	rt	News	Ι	Care	eers		Search		Caler	ndar		
	Title Job	: Num	ber:	Se 12	ervo ( 24	Cont	rol Er	ngi	nee	r							
	RC Ir cont next	ntern rol er gene	ational ngineei eration	l, a r to pro	leadin join o oducts	ig su iur te :. Re	ipplier eam. spons	of The ibili	con e cai ities	sume ndida for t	er ate thi:	level r • will jı s posi	radio oin ( tion	o con our hi will ii	troll ighly nclu	led vi y skill ide:	ehicle ed te

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 323). 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/CMS400Example/

index.asp?id=1&LangType=1036>, the following QueryString calls
return these values.

QueryString Call	Return value
Request.QueryString("id")	1
Request.QueryString("LangType")	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.

<% ecmContentBlockEx 13, "", 1 %>

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 XML Configuration 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
The default XSLT specified in the Edit XML configuration screen (illustrated below). Note that, in this example, the <b>XSLT Packaged</b> option is the default XSLT since it is selected.		none
Edit XML Configuration "new configuration"		
Fitle: new configuration		
ID:		
Description: xslt 2		
Display Information: (Files prefixed with/CMS300SampleTest/XmlFile		
Default:		
C XSLT 1: /xmlfiles/xslt/samplexslt1.xsl		
Name     xslt 2:     /xmlfiles/xslt/samplexslt2.xsl		
О XSLT 3:		

To use this display XSLT	Enter this for argument 2	Enter this for argument 3
<b>XSLT Packaged:</b> from the Edit XML configuration screen (that is, the XSLT created in the Data Designer)		0
Note: If an XSLT package has not been created, and zero (0) is the third argument, the default XSLT specified in the Edit XML configuration screen is used.		
XSLT 1 from the Edit XML configuration screen		1
XSLT 2 from the Edit XML configuration screen		2
XSLT 3 from the Edit XML configuration screen		3
An absolute or relative path to an XSLT	An external XSLT file not specified in the Edit XML Configuration screen. For example <% ecmContentBl ockEx 13, "samplexslt. xsl"%>	does not matter - if a value exists in argument 2, argument 3 is ignored

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

```
<</td><</td><% if request.QueryString("id")<> "" thenecmContentBlock(request.QueryString("id"))ecmContentBlock(14) end if %><</td>
```

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 comment.	
	ContentHtml,	
	SummaryHTML,	
	ContentLanguage,	
	FolderID,	
	GoLive,	
	EndDate,	
	MetaInfo	%>

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.

Attribute	Description	
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.	
FolderID	ID number of the content folder where the content block will be added to.	
GoLive	Date and time the content will go live.	
	Must match date/time format set in setup.asp.	
EndDate	Date and time the content will expire.	
	Must match date/time format set in setup.asp.	
MetaInfo	Specify the information for the metadata which will be used as a reference point for finding other words or information.	

## Add Content Example

The following is an example of the add content function in a template:.

```
<</td><% AddContent "admin", "admin", "", "Content Title", "Comment",</td>"This is the <b>body</b> of the content.","This is the <i>summary</i> for the content.", 1033, 0, "", "", "" %>
```

The previous example has the following properties:

- The user "admin" is used to be authenticated.
- The title of the new content block is "Control 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.
- No start date is specified.
- No end date is specified.
- No metadata is specified.

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"				
D 🗅 🗊 🔓 🕀 🐍	🐴 🚰 🗲   View: English (U.S.) 🗸   Add: -select language- 🗸 🭳			
Properties Content	Summary Meta Data Alias Comment Tasks			
Departies				
Content Title:	Content Title			
Content The.	381			
Content ID:				
Content Language	Annual (Dublished)			
Status:	Approved (Published)			
Last User To Edit:	Application Administrator			
Last Edit Date:	8/3/2005 12:27:47 PM			
Start Date:	[None Specified]			
End Date:	[None Specified]			
Action on End Date	[None Specified]			
Date Created:	8/3/2005 12:27:44 PM			
Approval Method:	Do Not Force All Approvers			
Approvals:	[None Specified]			
XML Configuration:	[None Specified] HTML Content Assumed(inherited)			
Path	X			
		*		
	Second Se			

# **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

<% ecmSearch or ecmSearchNoFormTag "Starting Folder", **Recursive.** "Target Page", Text Box Size, Maximum Characters, **"Button Image** Source", "Button Text", "Font Face", "Font Color". "Font Size", Horizontal, %> "Spare"

When using the search function, ecmSearch and ecmSearchNoFormTag can be interchanged. Shown here is the format for the ecmSearch and ecmSearchNoFormTag:

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	"Starting Folder" on page 337
Recursive	Whether to search subfolders. 0=No, 1 = Yes	Yes	"Recursive" on page 337
Target Page	The Web page to output the search results.	Yes	"Target Page" on page 338
Text Box	The size of the text box for user input.	Yes	"Text Box" on page 338
Attribute	Description	Required ?	For more information, see
---------------------	--	---------------	---
Maximum Characters	The maximum number of characters that the input text box can accept.	No	"Maximum Characters" on page 339
Button Image Source	The image location for the search button. Blank ("") = standard submit button.	No	"Button Image Source" on page 339
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	"Button Text" on page 340
Font Face	The font of the search tag text.	No	"Font Face" on page 341
Font Color	The font color of the search tag text. Blank ("") = Black.	No	
Font Size	The font size of the search tag text.	No	"Font Size" on page 342
Horizontal	Determines if the search options are in a row or a column. 1 = horizontal, 0 = vertical	Yes	"Horizontal" on page 343
Spare1	Currently not used.	Must be ""	"Spare 1" on page 343

## Search Tag Example

The following example places a text field and a search button in a table cell inside an ASP template.

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:



## 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
"1"	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	<ul> <li>Search</li> <li>O All the words</li> <li>O Any of the words</li> <li>O Exact Phrase</li> <li>□ Match Partial Words</li> </ul>

Text Box Size	Web Page View
50	Search All the words Any of the words Exact Phrase Match Partial Words

### **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 All the words Any of the words Exact Phrase Match Partial Words

Button Image Source	Web Page View
"search.jpg"	Search C All the words Any of the words Exact Phrase Match Partial Words
"http://www.ektron.com/ images/search.jpg"	Search C All the words Any of the words Exact Phrase Match Partial Words

### **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 All the words Any of the words Exact Phrase Match Partial Words

Button Text	Web Page View
"Click Me to Search"	<ul> <li>Seanch</li> <li>○ All Click Me to Search</li> <li>○ Any of the words</li> <li>○ Exact Phrase</li> <li>□ Match Partial Words</li> </ul>

### **Font Face**

This font is used for the search options displayed on the Web page.

Font Face	Web Page View
Cooper Black	Search © All the words © Any of the words © Exact Phrase Match Partial Words
Kristen ITC	Search © All the words © Any of the words © Exact Phrase □ Match Partial Words

### **Font Color**

The color of the text in the search tag. You may enter the color name or a hexidecimal value of the color.

Font Color	Web Page View
Red	Search All the words Any of the words Exact Phrase Match Partial Words
#14DA14	Search All the words Any of the words Exact Phrase Match Partial Words

### **Font Size**

This is the size of the text on the search tag.

Font Size	Web Page View
1	Search C All the words C Any of the words C Exact Phrase Match Partial Words

Font Size	Web Page View
5	© All the words ○ Any of the words ○ Exact Phrase □ Match Partial Words

### Horizontal

This attribute determines whether the search options are displayed vertically or horizontally.

Attribute	Web Page View
0 (vertical)	Search © All the words © Any of the words © Exact Phrase □ Match Partial Words
1 (horizontal)	Search  O All 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	%>

Attribute	Description	Required	For more information, see
Max Number of content blocks Returned	Maximum number of content blocks returned. (0=Unlimited)	Yes	"Max Number of Content Blocks Returned" on page 347
Style Info	An HTML style string used within the HTML "hyperlink" generated by the search output. Can control font, color, size, etc.	No	"Style Info" on page 348
ShowDate	Displays the last modified date of the content block. 0=No 1=Yes	No	"Show Date" on page 348

### 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.

<%	<pre>ecmSearchDisplay 0,"","1" %&gt;</pre>

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.

When the tag is added to a template, and a search is performed, the results appear similar to the following.



### 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:

## 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	Search Results
	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
3	Search Results
	RC Sportster RC Redstar RC Lilly

### **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
	Search Results 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
"font-family:arial; font- weight:bold; background- color:#cccccc; border:solid blue 1pt; margin:2px; width:50%;"	Search Results 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

#### **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.

Web Page View
Search Results
RC Sportster RC Redstar RC Lilly RC Cheetah
RC International Ships RC Sportster RC International Hosts RC Conference
RC International Announces the RC Cheetah
Search Results
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)

# **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

Attribute	Description	Options
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"
ShowInfo	Information that will appear to the right of the hyperlink.	"DateModified" "DateCreated" "AuthorFname" "AuthorLname"
MaxNumber	Maximum number of summaries returned.	0 = Unlimited
ObjType	Determines whether the list summary is for the folder or a single piece of content. For example: If the folder property = \books and "Folder" is chosen as the ObjType, the list summary for the folder with books appears. If the folder property = 1 and you choose "Content" as the ObjType, a single summary for content with an id = 1 appears. See Also: "Single Summary Function" on page 364 The default ObjType is "Folder" when left blank.	"" "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>

    <TD>

    <% ecmListSummary "\Marketing\News",0,1,"",</td>

    "Title,asc","DateModified",0,"","" %>

    </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.

#### 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 flagship product, the RC 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 the 3rd annual RC conference in Atlanta, Georgia.

## 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	RC International Ships RC Sportster RC International Hosts RC Conference RC International Announces the RC Cheet		
1	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 flagship product, the RC 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 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	
""	RC International Announces the RC Cheetah (8/27/2002 2:43:01 RC International, a leading RC racing company, has announced th availability of its new flagship product, the RC 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 the 3rd annual RC conference Atlanta, Georgia.	

Style Info	Web Page View	
"font-family:arial; font-weight:bold; background- color:#cccccc; border:solid blue 1pt; margin:2px;	RC International Announces the RC Cheetah RC International, a leading RC racing company, has announced tl availability of its new flagship product, the RC Cheetah.	
width:100%;"	RC International Hosts RC Conference	
	RC International, the fastest growing company in the RC industry, announces plans to host the 3rd annual RC conference Atlanta, Georgia.	

### 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
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.	, RC Sportster. The RC Sportster is design mid level RC racer currently on the market inference (8/27/2002 2:44:44 PM) growing company in the PC host the 3rd annual RC conference in Atla
***DateCreate d	Displays the date when the content was first created.	I level RC racer currently on the mark erence (4/25/2002 4:42:55 PM) owing company in the RC ost the 3rd annual RC conference in A
AuthorFname	Displays the first name of the user who last edited the content block.	mid level RC racer currently nference <i>(John)</i> growing company in the Ri host the 3rd annual RC co

Value	Description	Web Page View
AuthorLname	Displays the last name of the user who last edited the content block.	CT, KC Sportster. The KC Sp d mid level RC racer currentl Conference (Doe) st growing company in the R to host the 3rd annual RC co

### **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: "Starting Folder" on page 353	Any folder
Recursive	Allow the tag to get summary info from the child folders. See Also: "Recursive" on page 353	0 = Not Recursive 1 = Recursive
OrderBy, ASC/ DESC	Order the hyperlinks by: See Also: "Order By" on page 355	"Title" "DateModified" "DateCreated" "Startdate" "AuthorFname" "AuthorLname"
	Order the hyperlinks in ascending or descending order.	"Title,asc" "Title,desc"

Attribute	Description	Options
MaxNumber	Maximum number of summaries returned. See Also: "Max Number" on page 357	0 (Zero) = Unlimited
ObjType	Determines whether the list summary is for the folder or a single piece of content. For example: If the folder property = \books and "Folder" is chosen as the ObjType, the list summary for the folder with books appears. If the folder property = 1 and you choose "Content" as the ObjType, a single summary for content with an id = 1 appears. See Also: "Single Summary Function" on page 364 The default ObjType is "Folder" when left blank.	"" "Folder" "Content"

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	
<content></content>	Declares a new section of content in the XML collection.	
<id></id>	The content block ID. For example: <id>27</id>	
<title></title>	The title of the content block. For example: <title>HTML for the World Wide Web with XHTML and CSS</title>	
<quicklink></quicklink>	The quicklink associated with the content block. For example: <quicklink>/CMS400Example/ collection.aspx?id=27</quicklink>	
<teaser></teaser>	The summary of the content block. For example: <teaser>The easiest HTML book weve seen that still manages to be comprehensive.</teaser>	

XML Tags	Description
<startdate></startdate>	The content block's start date formatted as a .NET date type For example: <startdate>1/1/0001 12:00:00 AM</startdate>
<datemodified></datemodified>	The last date the content block was modified. For example: <datemodified>1/19/2005 4:18:06 PM<!--<br-->DateModified&gt;</datemodified>
<enddate></enddate>	The date the content block stops running on the Web site For example: <enddate>12/31/9999 12:00:00 AM</enddate>
<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>
<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. For example: <language>1033</language>
<displaydatemodified><!--<br-->DisplayDateModified&gt;</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<!--<br-->DisplayDateModified&gt;</displaydatemodified>

XML Tags	Description
<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<!--<br-->DisplayEndDate&gt;</displayenddate>

Here is how the raw XML list summary output appears before an XSLT is applied.

<content></content>
<id>24</id>
<title>ASP.NET Unleashed</title>
<quicklink>/CMS400Example/collection.aspx?id=24</quicklink>
<teaser><em>ASP.NET Unleashed, Second Edition</em> is really big, really thorough, and</teaser>
really <i>good</i> .
<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>/CMS400Example/collection.aspx?id=25</quicklink>
<teaser>Teach yourself how to write high-performance Web applications with ASP.NET and</teaser>
Visual C# .NETone step at a time.
<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>

# **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 350 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"	%>
	Sparer	/0-2

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>

    <TD>

    <% ecmSingleSummary "14",1,"","DateModified","" %>

    </TD>
```

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
	Servo Control Engineer #124 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%;	Servo Control Engineer #124 RC International is looked for a highly skilled servo control engineer to join our team.

### **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.	looked for a highly skilled se
DateModified	date the content was last modified.	ed for a highly skilled serve control am.
DateCreated	date when the content was first created.	#124 (4/25/2002 4:54:42 PM) ed for a bigbly skilled serve control
AuthorFname	first name of the user who last edited the content block.	er #124 (Mike) oked for a highry skilled ser
AuthorLname	last name of the user who last edited the content block	eer +124 <i>(Johnson)</i> ooked for a highly skilled serv

### 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	C 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>
</TD>
</TC>
</TC>
</TC>
```

The previous example has the following properties:

- Displays summaries from the \Books 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 a 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>
        <description></description>
        <item>
            <title>Visual Basic.NET How to Program</title>
            k>http://localhost/CMS400Example/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>
            k>http://localhost/CMS400Example/collection.aspx?id=26</link>
            <description>An Application-Driven Tutorial Approachp</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>
            k>http://localhost/CMS400Example/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>
            k>http://localhost/CMS400Example/collection.aspx?id=27</link>
            <description>The easiest HTML book weve seen that still manages to be comprehensive, and to
stay firmly grounded in the reality of todays demanding users and complex browsers.</description>
            <author>Application Administrator</author>
            <pubDate>1/19/2005 4:18:06 PM</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

Value	Summaries sorted by
DateModified	date modified
Date Created	date created
Startdate	The Start Date assigned to the content block when created.
AuthorFname	author's first names
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.

#### **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 ecmCollection 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. For

information about creating your own custom display functions, see "Creating Custom Display Functions" on page 380.

### 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.

Here is how a collection might appear on a Web page.



Below is the source code for the ecmNavigation display function.

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. See "Creating Custom Display Functions" on page 380 for more information.

#### 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.

<% ecmCollection 2, "ecmTeaser" %>

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.



#### **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. 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 = "
           width=""100%"">"
   for each info in cInfo
     html = html & "<a href="""</pre>
     html = html & CStr(info("ContentLinks"))
     html = html & """ >"
     html = html & Cstr(info("ContentTitle"))
     html = html & "</a>&nbsp;("
     html = html & info("DateModified")
     html = html & ")"
     html = html & Cstr(info("ContentTeaser"))
     html = html & " "
  next
  html = html & ""
   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

See "Creating Custom Display Functions" on page 380 for more information.

## **Collection Attributes**

### ID

The collection ID number is assigned to the collection when it is created.

View Collection "Intranet Press"			
수 🗕 읽 🗋 🗊	← English (U.S.) v 🗘		
Title			
Ektron Annound	es Winner of "All-Stars" Customer Competition		
Ektron Expands	Presence with Marketing, Interactive and Web		
Ektron Introduc	es an Enhanced Workflow Suite		
Ektron Offers a Visual Development Environment for Rapid CMS			
Ektron Supports Rapid and Efficient Globalization Strategies on			
Ektron, Inc. Named One of New England's Fastest Growing Tech			
eWebEditPro+XML V4.2 supports the vision of create content or			
<u>More info</u>			
Title:	Intranet Press		
ID:	6		
Template:	/CMS400Example/intranet/pr.aspx		
Last User To Edit: Application Administrator			
Last Edit Date: 1/19/2005 5:06:53 PM			

### **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

## **Creating Custom Display Functions**

The following section explains how to create custom display functions for your ecmCollection function.

### **Function Structure**

All display functions use a similar structure:

```
Function yourCollectionDisplayFunction(cInfo)
dim html, info
html = html & "<b>List of Content Titles:</b><br>"
for each info in cInfo
    html = html & info("ContentTitle") & "<br>"
next
yourCollectionDisplayFunction = html
end Function
```

#### Creating a Basic Display Function

Explained below is a how to create a basic display function.

1.	In the API file, define the display function.	Function MyDisplayFunction(cInfo)
		end Function
2.	Declare any necessary attributes	Function MyDisplayFunction(cInfo) dim html, info
		end Function
3.	Add any html or text to apply to the collection data as a whole, not individually (for example, add a opening and closing html table tag)	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>"</ul></pre>
		ntmi = ntmi & "" end Function

<ol> <li>Add a for loop to cycle through each collection item, and apply the information defined</li> </ol>	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo next html = html &amp; "</ul>" end Function</pre>
<ol> <li>Inside the for loop, create the string to be applied to each item generated by the collection data.</li> </ol>	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo</ul></pre>
For information about the building blocks, see "Building Blocks" on page 383.	<pre>next html = html &amp; "" end Function</pre>
This places an <li> tag at the beginning and end of each item .</li>	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo html = html &amp; "<li>" html = html &amp; "</li>" next html = html &amp; "</ul>" end Function</pre>
This creates the hyperlink for the collection item. When the collection item is generated on the Web page, the hyperlink sends the visitor to the corresponding content block.	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo html = html &amp; "<li>" html = html &amp; "<a "="" &="" contentlinks")="" href=" &amp; info(">" &amp; Cstr(info("ContentTitle")) &amp; "</a>" html = html &amp; "</li>" next html = html &amp; "</ul>" end Function</pre>

This displays the summary of the content block in the collection.	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo html = html &amp; "<li>" html = html &amp; "<a "="" &="" contentlinks")="" href=" &amp;&lt;br&gt;info(">" &amp; Cstr(info("ContentTitle")) &amp; "</a>" html = html &amp; Cstr(info("ContentTeaser")) html = html &amp; "</li>" next html = html &amp; "</ul>"</pre>
<ol> <li>Add this line at the end of the function to return the HTML to be output to the ecmCollection function.</li> </ol>	<pre>Function MyDisplayFunction(cInfo) dim html, info html = "<ul>" for each info in cInfo html = html &amp; "<li>" html = html &amp; "<a "="" &="" contentlinks")="" href=" &amp;&lt;br&gt;info(">" &amp; Cstr(info("ContentTitle")) &amp; "</a>" html = html &amp; Cstr(info("ContentTeaser")) html = html &amp; "</li>" next html = html &amp; "</ul>" MyDisplayFunction = html end Function</pre>

### **Building Blocks**

The sample display function created above uses only two building blocks. This section explains all building blocks you can use within a custom display function.

Name	Description	Web Page View
info("ContentTeaser")	Displays content block summary	<ul> <li>Plastic Molder #123 RC International is looking for an experienced plastics molder.</li> <li>Servo Control Engineer #124 RC International is looked for a highly skilled servo control engineer to join our team.</li> </ul>
info("Comment")	Displays content block comment	<ul> <li>Plastic Molder #123         This is the historical comment for Plastics Mi     </li> <li>Servo Control Engineer #124         This is the historical comment for Servo Con     </li> </ul>
info("ContentID")	Displays content block ID number	<ul> <li>Plastic Molder #123 ID= 13</li> <li>Servo Control Engineer #124 ID= 14</li> </ul>
info("ContentLinks")	Displays content block path relative to your site's root	<ul> <li>Plastic Molder #123 /CMS400Sample/hr.asp?id=13</li> <li>Servo Control Engineer #124 /CMS400Sample/hr.asp?id=14</li> </ul>

Name	Description	Web Page View
info("ContentTitle")	Displays content block title.	<ul> <li>Plastic Molder #123 /CMS400Sample/hr.asp?id=13</li> <li>Servo Control Engineer #124 /CMS400Sample/hr.asp?id=14</li> </ul>
info("DateModified")	Displays date and time content block was last modified.	<ul> <li>Plastic Molder #123 8/27/2002 2:38:34 PM</li> <li>Servo Control Engineer #124 8/27/2002 2:40:26 PM</li> </ul>
info("DisplayDateMo dified")	Displays date and time content block was last modified using the date/ time format specified in Setup.asp.	<ul> <li>Plastic Molder #123 27-Aug-2002 02:38:34 PM</li> <li>Servo Control Engineer #124 27-Aug-2002 02:40:26 PM</li> </ul>
info("EndDate")	End date and time for the content block	
info("DisplayEndDat e")	Displays end date and time for content block using date/time format specified in Setup.asp.	
info("GoLive")	Start date and time for content block	<ul> <li>Plastic Molder #123 This content went live on: 10/30/2002 10:</li> <li>Servo Control Engineer #124 This content went live on: 10/29/2002 6:5</li> </ul>

Name	Description	Web Page View	
info("DisplayGoLive ")	Displays start date and time for the content block using the date/ time format specified in Setup.asp.	<ul> <li>Plastic Molder #123 This content went live on: 30-Oct-2002 10</li> <li>Servo Control Engineer #124 This content went live on: 29-Oct-2002 06</li> </ul>	
info("EditorFname")	First name of user who last edited content block.	<ul> <li>Plastic Molder #123 Application</li> <li>Servo Control Engineer #124 Application</li> </ul>	
info("EditorLname")	Last name of user who last edited content block.	<ul> <li>Plastic Molder #123 Administrator</li> <li>Servo Control Engineer #124 Administrator</li> </ul>	

### **Defining Display Functions**

After you create a display function in the API, define it in the ecmCollection function as a new case. To do so, follow these steps.

- 1. Locate the ecmCollection function in the API file.
- 2. Find the line that contains Select Case displayFunction.
- 3. Under that line, insert the following, replacing MyDisplayFunction with the function you created.

```
Case "MyDisplayFunction"
html = MyDisplayFunction(cInfo)
```

4. Save and close the file.

With the display function defined, and added to the ecmCollection function as a new case, you are ready to use the display function to define how collections look.

#### **Using a Custom Display Function**

To use a custom display function, follow these steps.

- 1. Add the ecmCollection function to a template.
- 2. Specify the ID of the collection to be displayed.
- 3. Add the display function to define the collection's appearance.

<% ecmCollection "2", "MyDisplayFunction" %>

When displayed on the Web site, the collection looks like the following.

Plastic Molder #123

RC International is looking for an experienced plastics molder.

Servo Control Engineer #124

RC International is looked for a highly skilled servo control engineer to join our team.

# **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 a 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>
        <description></description>
        <item>
            <title>Visual Basic.NET How to Program</title>
            k>http://localhost/CMS400Example/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>
            k>http://localhost/CMS400Example/collection.aspx?id=26</link>
            <description>An Application-Driven Tutorial Approachp</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>
            k>http://localhost/CMS400Example/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>
            k>http://localhost/CMS400Example/collection.aspx?id=27</link>
            <description>The easiest HTML book weve seen that still manages to be comprehensive,
and to stay firmly grounded in the reality of todays 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	Determines whether the HTML contained within the content is included in the XML output. 0 = Do not get HTML 1 = Get HTML
	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.

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>	The content block ID. For example: <id>27</id>	
<title></title>	The title of the content block. For example: <title>HTML for the World Wide Web with XHTML and CSS</title>	
<quicklink></quicklink>	The quicklink associated with the content block. For example: <quicklink>/CMS400Example/ collection.aspx?id=27</quicklink>	
<teaser></teaser>	The summary of the content block. For example: <teaser>The easiest HTML book weve seen that still manages to be comprehensive.</teaser>	
<html></html>	All of the HTML content contained in a content block. See Also: "GetHtml" on page 389 For example: <html><em>ASP.NET Unleashed, Second Edition<!--/<br-->em&gt; is really big, really thorough, and really <i>good</i>. 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. </em></html>	
<startdate></startdate>	The content block's start date formatted as a .NET date type For example: <startdate>1/1/0001 12:00:00 AM</startdate>	

XML Tags	Description
<datemodified></datemodified>	The last date the content block was modified. For example: <datemodified>1/19/2005 4:18:06 PM<!--<br-->DateModified&gt;</datemodified>
<enddate></enddate>	The date the content block stops running on the Web site For example: <enddate>12/31/9999 12:00:00 AM</enddate>
<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="/CMS400Example/&lt;br&gt;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><!--<br-->DisplayDateModified&gt;</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<!--<br-->DisplayDateModified&gt;</displaydatemodified>

XML Tags	Description
<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<!--<br-->DisplayEndDate&gt;</displayenddate>
<enddateaction></enddateaction>	<pre>What should happen to the content block on its end date.     archive display     archive expire     refresh report For example:     <enddateaction>Archive_Expire</enddateaction></pre>
<comment></comment>	Comments contained in the content block. For example: <comment>move picture below text</comment>

Here is how the raw XML Collection output appears before an XSLT is applied.

<content></content>
<id>24</id>
<title>ASP.NET Unleashed</title>
<quicklink>/CMS400Example/collection.aspx?id=24</quicklink>
<teaser><em>ASP.NET Unleashed, Second Edition</em> is really big, really thorough, and</teaser>
really <i>good</i> .
<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><a href="/CMS400Example/collection.aspx?id=24">ASP.NET Unleashed</a><!--<br-->Hyperlink&gt;</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>/CMS400Example/collection.aspx?id=25</quicklink>
<teaser>Teach yourself how to write high-performance Web applications with ASP.NET and Visual C# .NETone step at a time.</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><a href="/CMS400Example/collection.aspx?id=25">Microsoft ASP.NET Programming</a></hyperlink>
with Microsoft Visual C#.NET
<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	
<u>More info</u>	Drodusts
Title:	Products 1
<u>More info</u> Title: ID: Template:	Dreducts 1 /CHISSESSample/products.asp
<u>More info</u> Titlo: ID: Tomplato: Last User To Edit:	Products 1 /CMOCCCCample/products.asp : Application Administrator
<u>More info</u> Title: ID: Template: Last User To Edit: Last Edit Date:	Droducts 1 /CHISSESSample/products.asp Application Administrator 29-Oct-2002 09:36 PM
More info Title: ID: Tomplate: Last User To Edit: Last Edit Date: Date Created:	Products 1 /CMOCCCCample/products.asp Application Administrator 29-Oct-2002 09:36 PM 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.



# **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" %>

Νοτε

You can use ecmCalendar to call a calendar and display a different view, such as daily. For more Information, see "Calendar Function" on page 399

The "**id**" argument represents the ID number of the event calendar in the Workarea.



Table Properties:

### **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, \CMS400example\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.

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

lf	Calendar Display				
Logged in					
	<< May		June, 2005		July
	Monday	Tuesday	Wednesday	Thursday	Friday
			To 🔍 1	<b>2</b>	<b>N</b>
	🔁 🔍 🛛 6	T C 7	To Q 8	P 9	🔁 🔍 🔹 🕐
			Centaur's Online Marketing 2005	Centaur's Online Marketing 2005	
	To 🔍 13	To Q 14	To Q 15	To 🔍 16	· 🔁 🔍 🔹 🕐
		Internet World			
	20	21	<u>.</u> 22	23	🖪 🔍 👘
	<b>Q</b> 27	<b>1 2 2 8</b>	Te Q 29	Te 🔍 30	

lf	Calendar Display				
Logged Out					
	< May		June, 2005		July >>
	Monday	Tuesday	Wednesday	Thursday	Friday
			1	2	3
	6	7	8	9	10
			Centaur's Online	Centaur's Online	
			Marketing 2005	Marketing 2005	
	13	14	1	5 16	17
		Internet World			
	20	21	2	2 23	24
	27	28	2	9 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 396

The ecmCalendar function attributes are explained below.

Attribute	Description
id	The " <b>id</b> " argument represents the ID number of the event calendar in the Workarea.
	View Calendar "Calendar Example"
	🛐 📝 📃 🗊 🔶 View In: English (U.S.) 👻 Add In
	Title: Calendar Example ID 1 Description
	Location Location:
	Start Time Start:
	End Time End:
	Table Properties:

Attribute	Description		
DisplayType	<ul><li>Specify the Calendar display type.</li><li>Month - A 30 day calendar month view.</li></ul>		
	Noticity       Mandee       Testerse       Here ster       Fore ster       Fore ster       Exter ster         1       Mandee       1		
	• Day - Single day view.		
	<ul> <li>Inline - Display events that occur from the start date of the calendar</li> </ul>		
	through the end date.  Sunday, May 01, 2005 - Saturday, July 30, 2005 S-503035 Hockey game Win Sub 700 PM End 10:30 PM Sub 700 PM End 40:001 Basketsal Memasial High School Bit		
	<ul> <li>MonthUpWithEvent - Displays a month calendar. Events appear to the right of the calendar when you click on a day.</li> <li> <sup>Sum Mon Tue Wed Thu Tri Set Hockey game</sup>         1 2 3 4 5 6 7 Start.7:00 PM         1 1 12 13 14 End:10:30 PM         15 16 17 18 19 20 21         200420052006     </li> </ul>		

### Calendar Style Sheet

Before you insert an event Calendar into your Ektron CMS400.NET template, you should make a reference to the style sheet, \CMS400example\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.

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

lf	Calendar Display									
Logged in	Notice the add event a	And view       May       Image: state sta	v even vee, 20 Veed 1 8 15 22 29 2005.	t butte 05Ju 7hu 2 9 16 23 30 . 2000	Sons f Fri 3 10 17 24 5	that appear when logged in. 6/9/2005 Field Hockey Location:High School Start:5:00 PM End:8:00 PM				

lf	Calendar Display										
Logged Out											
	<u>&lt;&lt; M</u>	ay Ju	ne, 20	)05 <u>Ju</u>	<u> </u> >>	6/9/2005					
	Mon	Tue	Wed	Thu	Fri	Field Hockey					
			1	2	3	Start:5:00 PM					
	6	7	8	9	10	End:8:00 PM					
	13	14	15	16	17						
	20	21	22	23	24						
	27	28	29	30							
	20	<u>)04</u> :	2005.	. <u>200</u>	<u>6</u>						

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 406.

The "id" argument represents the form's ID number, visible when viewing the form from the View Form screen.

View Form "Email Contact Form"								
🔰 🖻 🏹 🔚 🍓 🐴 🛍 💣 🗲   View: English (U.S.) 🛛 🗸								
Form Properties Form	Post back message							
Form Title:	Email Contact Form							
FormID:	259							
Status:	Approved (Published)							
Description:								

ecmFormBlock and ecmFormBlockNoFormTag are interchangeable when using this function. Here is an example of a form content block in an Ektron CMS400.NET template.

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 324.

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: "Creating an Item List" on page 407 and "Item List" on page 412	No
Spare1	Not yet implemented See Also: "Spare 1" on page 414	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

Attribute	Description	Required?
; (semicolon)	Separator: Separates content block id from the exclusion list	No ( <b>Yes</b> if including an <b>exclusion</b> 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	Туре	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;
</pre>
```

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>
<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: "Creating an Item List" on page 407

Some examples might be:

```
<% ecmMetaData "[1][2]","" %>
```

🖉 metadata[1] - Notepad	_ 🗆 🗙
<u>File E</u> dit F <u>o</u> rmat <u>H</u> elp	
<pre></pre>	•
 <body> </body> 	T

In this example, all metadata from content block ID=1 and 2 is displayed.

```
<% ecmMetaData "[1;title][2;keywords]","" %>
```

┛ m	etad	ata[1] -	Notepad		- 🗆 ×
Eile	Edit	Format	Help		
					<b></b>
DC<br Trar <htm <hea <tit <met mana</met </tit </hea </htm 	DCTY nsit nl> ad> tle> ta n agem	PE HTM ional, CMS40( ame="Ment; s	L PUBLIC "-//W3C//D /EN"> Login Information< eywords" content="C olution; business u	TD HTML 4.0 /Title> MS400; content sers; developers;	
	ead> dy>	5 >			
<td>ody&gt; cml&gt;</td> <td> </td> <td></td> <td></td> <td>•</td>	ody> cml>				•

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	_ 🗆 🗙
<u>File E</u> dit F <u>o</u> rmat <u>H</u> elp	
<pre><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">     <html> <html></html></html></pre>	•
<pre><title>CMS400 Sample web Site</title> <meta content="CMS400; content management; solution; business users; develop benefits" name="Keywords"/></pre>	t ers;
 <body></body>	_
k/body> 	
	-

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	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:
	<%=ecmShowSiteLanguage(`')%>
	Or you can pass in an argument such as:
	<%=ecmShowSiteLanguage(`mysitelanguage')%>

# **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.

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,'<img src=graphic file name>')%>

#### For example:

<%= ecmDHTML\_DropNextGenMenu(1,'<img src=mymenugraphic.jpg>')%>

#### **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 418

## 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/CMS400Example/*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
fontFamily	Font style of menu text See Also: "FontFamily" on page 419	arial,helvetica,espy, sans-serif
fontColor	Color of menu text	#000000
	Note: When changing the color, you can specify the color name or code.	
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
menultemBorder	Width (in pixels) of border around each menu item	1
menultemBgColor	Background color of menu items	#6699ff
menuLiteBgColor	Color of main menu border highlight See Also: "menuLiteBgColor" on page 419	#ffffff
menuBorderBgColor	Color of the main menu border	#6699ff
menuHiLiteBgColor	Background color of highlighted menu item	#000084

Attribute name	Specifies the	Default value
menuContainerBgColor	Background color of the container	#6699ff
	Note: The container is like a <div> tag that displays the menu. When you set the container color, it appears at the top and left side of the menu frame only.</div>	
childMenulcon	Path to the submenu icon (the image used to indicate that a submenu is available from this menu item). RC Cars RC Planes Droducts Page To display a different image, either provide an external path to it or upload the image to Ektron CMS400.NET then provide a local path.	./Workarea/ images/ application/ arrows.gif
childMenuIconHiLite	The path to the icon that appears when you hover the mouse over a menu item that leads to a submenu. To display a different image, either provide an external path to it or upload the image to <b>Ektron</b> <b>CMS400.NET</b> then provide a local path.	./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

Home Page Content	
Private Content	
Plane	
Yahoo	
Sub Menu1	
👌 Add	👌 Add
📝 Edit	📝 Edit

# 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
ld	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>/CMS400Example/collection.aspx?id=23</ItemLink>
                   <ItemTarget>2</ItemTarget>
               </Item>
           </Menu>
       </Item>
   </Menu>
</Ttem>
```

# **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/CMS400Example/WorkArea/WebServices/ ContentWS.asmx?WSDL" />

and

<cfset application.AppPath = "/CMS400Example/WorkArea/platforms/ColdFusion/" />

The application.WSDLpath needs to updated with the location of the WSDL file. For example, For example, if your site is www.myektronsite.com, the WSDLpath would be http://www.myektronsite.com/WorkArea/WebServices/ContentWS.asmx.

The application.AppPath needs to updated with the path to the ColdFusion custom Functions.

To customize you Ektron CMS400.NET application, you will need to be familiar with these tags

Function	Description	More Information
Calendar Functions		
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 441
Collection Functions		
Collection	Displays links generated by collections	"Collection Function" on page 435

Function	Description	More Information
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 440
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 429
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 429
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 441
Content Functions		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> progammatically.	"Add Content" on page 427
Content Block	Displays and manages a content block	"Content Block Function" on page 426
Form	Displays a content block associated with a form	"Form Function" on page 444
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 445
Language Functions		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 450
List Summary Functions		
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 434
List Summary	Displays a list of content block summaries	"List Summary Function" on page 431

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 432
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 433
Single Summary	Displays the summary for a content block.	"Single Summary Function" on page 434
Login Functions		
Login	Adds the login or logout button	"Login Function" on page 426
Menu Functions		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 448
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 448
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 449
Metadata Functions		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 445
Search Functions		
Search	Inserts a text box with a search button	"Search Function" on page 429
Search Display	Creates an area to display search results	"Search Display Function" on page 431

## 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 317 for details. The only difference is the syntax used to implement the function. This is listed below.

## Login Tag Example

<cfmodule template="#request.ecm.AppPath#ecmLogin.cfm">

# **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 ContenBlockEX function in ASP. See "Content Block Function" on page 319 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**

<CFMODULE id="1" template="#request.ecm.AppPath#ecmContentBlock.cfm">

## **Dynamic Content Block**

<CFMODULE id="#id#" template="#request.ecm.AppPath#ecmContentBlock.cfm">

### **XML Content Block**

<CFMODULE id="13" XSLT= "" override= "1"

template="#request.ecm.AppPath#ecmContentBlockEx.cfm">

# 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.

<cfmodule ,<br="" template="#request.ecm.AppPath#ecmAddContent.cfm" username="">Password="", Domain="", content_title="", content_commer ContentHtml="", SummaryHTML= ContentLanguag FolderID="", GoLive="", EndDate="", MetaInfo=""</cfmodule>	nt="", "", e="",
induind-	-

Here is the format for the add content function:

This function is basically the same as the Add Content function in ASP. See "Add Content" on page 329 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:.

## **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 394 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 395 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 333 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

>

```
<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">
```

## 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 344 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

>

<CFMODULE template="#request.ecm.AppPath#ecmSearchDisplay.cfm"

MaxNumber="0" StyleInfo="" ShowDate="1">

## 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 350 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>

</TR>

>

<TD>

```
<CFMODULE template="#request.ecm.AppPath#ecmListSummary.cfm"
Folder="\Marketing\News" MaxNumber="0" Recursive="0"
ShowSummary="1" StyleInfo="" OrderBy="Title,asc" ShowInfo="" >
</TD>
```

# 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 358 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 368 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>
</TD>
</TD>
</TD>
</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 431 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 364 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 374 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="cms0bj">
<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 & "<a href=""">
       <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>
       <cfelse>
           <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 & "</a>">
   </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.



#### **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. 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="">
<cfparam name="Attributes.target" default="">
<cfset ErrorString = "">
<cfset ErrorString = "">
```

```
<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>">
       <cfset tString = tString & "<a href=""">
       <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") & "</a>" & "&nbsp;(" &
       DateFormat(cT.Item("DateModified"), "mm/dd/yyyy") & " " &
       TimeFormat(cT.Item("DateModified"), "h:mm:ss tt") & ")" &
       cT.Item("ContentTeaser")>
       <cfset tString = tString & "</td>&nbsp;">
```

```
</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 379

#### **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.

#### **Building Blocks**

See "Building Blocks" on page 383

## 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 389 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 387 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 396 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	The "id" argument represents the ID number of the event calendar in the Workarea.	
	View Calendar "Calendar Example"	
	31 🕼 📰 🗂 🔶 View In: English (U.S.) 👻 Add In	
	Title: Calendar Example ID 1 Description	
	Location:	
	Start Time Start:	
	End Time End:	
	Table Properties:	
Attribute	Description	
-------------	---	
DisplayType	Specify the Calendar display type.	
	• Month - A 30 day calendar month view.	
	Noticity     Tarestrag     Weaklawer     Tarestrag     Veaklawer     Tarestrag     Tarest	
	Day - Single day view.	
	<ul> <li>Inline - Display events that occur from the start date of the calendar through the end date.</li> </ul>	
	Sunday, May 01, 2005 - Saturday, July 30, 2005 SIS/3005 Hotolog game WWN Start.700 PM End 512.30 PM Signapos Youth Basketboal Menasci Haja School Start.200 PM End 4 50 PM	
	<ul> <li>MonthUpWithEvent - Displays a month calendar. Events appear to the right of the calendar when you click on a day.</li> <li> <sup>Sun Mon Tue Wed Thu Fri Set Hockey game</sup>         1 2 3 4 5 6 7 Start.7:00 PM         1 1 12 13 14 End:10:30 PM         15 16 17 18 19 20 21         200420052006     </li> </ul>	

Here is a sample of the Event Calendar syntax.

### **Event Calendar Sample**

```
<CFModule template="#request.ecm.AppPath#ecmEvtCalendar.cfm" calendar_id="24"
DisplayType="monthupwithevent">
```

# Form Function

	The ecmFormBlock function displays an HTML form content block. For more information about HTML forms, see the <b>Ektron</b> <b>CMS400.NET</b> 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 id="" template="#request.ecm.AppPath#ecmFormBlock.cfm"></cfmodule>
	This function is basically the same as the Form function in ASP. See "Form Function" on page 403 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.

<CFModule template="#request.ecm.AppPath#ecmFormBlock" form\_id="1">

# **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/CMS400Example/index.asp?id=2

To pass in a Form content block, pass the ekfrm value in the url parameter, as illustrated below.

http://localhost/CMS400Example/index.asp?ekfrm=2

**NOTE** For more information about passing the id value as a url parameter, see "Dynamic Content Block" on page 427.

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 407 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]>

🛃 metadata[1] - Notepad	
<u>File Edit Format Help</u>	
HTML PUBLIC "-//W3C//DTD HTML 4.0<br Transitional//EN"> <html> <head> <title>CMS400 Login Information</title></head></html>	
<pre><metacm3400 in="" login="" onmatten()="" refer<br=""><meta content="CMS400; content&lt;br&gt;management; solution; business users; developers;&lt;br&gt;benefits" name="Keywords"/></metacm3400></pre>	
 <body></body>	_
 	•

<cfmodule template="#request.ecm.AppPath#ecmMetadata.cfm"

```
ItemList="[1][2;title,keywords]>
```

```
imetadata[2] - Notepad

implies Edit Format Help

implies Edit Format He
```

# **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= '<img src=graphic file name>'
template="#request.ecm.AppPath#ecmDHTML_DropNextGenMenu.cfm">
```

#### For example:

```
<CFMODULE MenuId="1" Title= '<img src=mymenugraphic.jpg>'
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/CMS400Example/* Workarea/java/. For a description of the ek\_menu.js file, see "ek\_Menu.js" on page 418.

### 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 421 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 415 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/CMS400Example/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/CMS400Example/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.myektronsite.com, the WSDLpath would be http://www.myektronsite.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
Calendar Functions		
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 468
Collection Functions	3	
Collection	Displays links generated by collections	"Collection Function" on page 462
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 466
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 467
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 468
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 467
Content Functions		
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> progammatically.	"Add Content" on page 456
Content Block	Displays and manages a content block	"Content Block Function" on page 455
Form	Displays a content block associated with a form	"Form Function" on page 471
Multipurpose	Displays either a content block or a form	"MultiPurpose Function" on page 472
List Summary Functions		

Function	Description	More Information
Archive ListSummary	Displays summaries of archived content blocks	"Archive List Summary Function" on page 460
List Summary	Displays a list of content block summaries	"List Summary Function" on page 459
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 461
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 462
Single Summary	Displays the summary for a content block	"Single Summary Function" on page 460
Login Functions		
Login	Adds the login or logout button	"Login Function" on page 454
Menu Functions		
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 475
GenerateMenu	Works with DropHTML menu to display DHTML menus	"DHTML Menus" on page 475
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 477
Metadata Functions		
Metadata	Displays the metadata for the current Web page	"Metadata Function" on page 473
Search Functions		
Search	Inserts a text box with a search button	"Search Function" on page 457

Function	Description	More Information
Search Display	Creates an area to display search results	"Search Display Function" on page 458
Language Functions		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 478

# Login Function

This function is responsible for displaying the following buttons on the Web page.

Button	Name	Description
🎉 Login	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	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	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
Preview	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.
Preview	Preview Off	Turns off preview mode.

This function is basically the same as the Login function in ASP. See "Login Function" on page 317 for details. The only difference is the syntax used to implement the function. This is listed below.

### Login Function Example

# **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 472.

This function is basically the same as the Content Block function in ASP. See "Content Block Function" on page 319 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**

<to< td=""><td>d&gt;</td><td></td><td></td></to<>	d>		
	php</td <td>ecmContentBlock(1)</td> <td>?&gt;</td>	ecmContentBlock(1)	?>
</td <td>td&gt;</td> <td></td> <td></td>	td>		

### **Dynamic Content Block**

### XML Content Block

# 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</th <th>("Username" "Password", "Domain", 'content_title", "content_comment", "ContentHtml", "SummaryHTML", "ContentLanguage", "FolderID", "GoLive", "EndDate",</th> <th></th>	("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 329 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:.

# **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.

?>

This function is basically the same as the Search function in ASP. See "Search Functions" on page 333 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,""); ?>
```

# 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 344 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

```
    <</td>

    <</td>
```

# 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.

This function is basically the same as the List Summary function in ASP. See "List Summary Function" on page 350 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>

<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 459 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",
```

```
"Sparel");
```

?>

This function is basically the same as the Single Summary function in ASP. See "Single Summary Function" on page 364 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>
</TD>
</TD>
</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 358 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 368 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("\Books\\", 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 374 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.



Below is the source code for the display function "ecmNavigation".

```
Function ecmNavigation($cInfo) {
  global $html, $info;
   $html = "
     width=\"100%\">";
   $mycount = $cInfo->count();
   for ($iloop=0; $iloop < $mycount; ++$iloop) {</pre>
      $info = $cInfo->Item($iloop+1);
      $html = $html."  <a href=\"";</pre>
      $html = $html.$info->Item("ContentLinks");
      $html = $html."\">";
      $html = $html.$info->Item("ContentTitle");
      $html = $html."</a>&nbsp;";
   }
  $html = $html."";
  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.



#### **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 = "";
  $mycount = $cInfo->count();
  for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item($iloop+1);
    $html = $html."*html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $info = $cInfo->Item("ContentTitle");
    $html = $html."*for ($iloop=0; $iloop < $mycount; ++$iloop) {
    $html = $html."</td>*for ($contentTeaser");
    $html = $html."*for ($contentTeaser");
    $html = $html."*for ($contentTeaser");
    $html = $html."
```

```
}
$html = $html."";
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 379

### **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.

#### **Building Blocks**

See "Building Blocks" on page 383

# 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 389 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 387 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 394 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 395 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 396 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	The " <b>id</b> " argument represents the ID number of the event calendar in the Workarea.
	View Calendar "Calendar Example"
	31 📝 📃 🗍 🔶 View In: English (U.S.) 👻 Add In
	Title: Calendar Example ID 1 Description
	Location Location:
	Start Time Start:
	End Time End:
	Table Properties:

Attribute	Description
DisplayType	Specify the Calendar display type.
	Monthline A door day calculation information.
	• Day - Single day view.
	<ul> <li>Inline - Display events that occur from the start date of the calendar through the end date.</li> </ul>
	Sunday, May 01, 2005 - Saturday, July 30, 2005 552985 Hockey game WWA Balanz 00 PM End 10.20 PM Sci2005 Youth Baskeball Newsist High Scinool Balanz 200 PM End 4 00 PM
	• <b>MonthUpWithEvent</b> - Displays a month calendar. Events appear to the right of the calendar when you click on a day.
	<         Apr         May, 2005         Jun >>         5/5/2005           Sun         Mon         Tue         Wed         Thu         Fri         Sat         Hockey game           1         2         3         4         5         6         7         Sat         Location: VWA           8         9         10         11         12         33         14         End.         End.         For Location: VWA           15         16         17         18         19         20         21         End.         10.30 PM           12         23         24         25         26         27         28           29         30         31         =         0         End.         10.30 PM

### **Event Calendar Sample**

Here is an example of the event Calendar in an Ektron CMS400.NET template.

</td <td>php ecmEvtCalendar(24, "MonthUpWithEvent");</td>	php ecmEvtCalendar(24, "MonthUpWithEvent");

# 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 403 for details.

The only difference between the PHP function and the ASP one is the syntax for implementing it. The PHP example is below.

```
<</td><</td>As an alternative, you can use the multipurpose function, which can display either
```

**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 472.

# **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/CMS400Example/index.asp?id=2

To pass in a Form content block, pass the ekfrm value in the url parameter, as illustrated below.

http://localhost/CMS400Example/index.asp?ekfrm=2

**NOTE** For more information about passing the id value as a url parameter, see "Dynamic Content Block" on page 456.

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", "Sparel";) ?>

This function is basically the same as the Metadata function in ASP. See "Metadata Function" on page 407 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;</pre>

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]","") ?>
```

🖉 metadata[1] - Notepad	_ 🗆 🗙
<u>File E</u> dit F <u>o</u> rmat <u>H</u> elp	
<pre><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">     <html> <html></html></html></pre>	<b></b>
<pre><title>CMS400 Sample web Site; CMS400 Login Information</title> <meta content="CMS400; content management; solution; business users; developers; benefits; Ektron; login" name="Keywords"/></pre>	
 <body></body>	
 	<b>•</b>

```
<?PHP ecmMetaData ("[1;title][2;keywords]","") ?>
```

```
ile Edit Format Help

ile Concent Help

ile Edit Format Help

ile Edit Format Help

ile Concent Help

ile Edit Help

ile Concent Help

</
```

<?PHP 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>CMS400 Sample Web Site</Title>
<meta name="Keywords" content="CMS400; content
management; solution; business users; developers;
benefits">
</head>
</body>
</head>
</body>
</head>
</body>
</head>
```

### 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,"MyMenul"; ?>

#### If the user clicks a graphic to invoke the menu, use this syntax:

<?php ecmDHTML\_DropNextGenMenu menu ID number, '<img src=graphic file name>'; ?>

#### For example:

<?php ecmDHTML\_DropNextGenMenu 1,'<img src=mymenugraphic.jpg>'; ?>

#### **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/CMS400Example/*Workarea/java/. For a description of the ek\_menu.js file, see "ek\_Menu.js" on page 418.

### 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 421 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.

Site Language:	English (US) 🗾 💌
	Danish
	English (US)
	French (Standard)
	German (Standard)

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 415 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/CMS400Example/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/CMS400Example/ 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.myektronsite.com, the WSSoap\_address would be http://www.myektronsite.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		
Calendar Functions				
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 dsiplayed, see "DisplayType" on page 400	"Calendar Function" on page 483		
Event Calendar	Displays a month view event calendar	"Event Calendar Function" on page 484		
Collection Functions	3			
Collection	Displays links generated by collections	"Collection Function" on page 484		
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 485		
Random Content	Randomly displays the content of a collection item	"Random Content Function" on page 486		
Random Summary	Randomly displays the summary of a collection item	"Random Summary Function" on page 486		
RSS Collection	Creates an RSS feed out of the items in a collection.	"RSS Collection Function" on page 487		
Content Functions				
Add Content	Allows a user to add content to <b>Ektron CMS400.NET</b> progammatically.	"Add Content" on page 487		
Content Block	Displays and manages a content block	"Content Block Function" on page 488		
Form	Displays a content block associated with a form	"Form Function" on page 489		
Language Functions	3			

Function	Description	More Information		
Language Select	Lets a site visitor select a language in which to view site.	"Language Select Function" on page 491		
Login Functions				
Login	Adds the login or logout button	"Login Function" on page 491		
Menu Functions				
DropHTMLMenu	Used to display a DHTML menu	"DHTML Menus" on page 492		
GenerateMenu	Works with DropHTML menu to display DHTML menus.	"DHTML Menus" on page 492		
XML Menu	Creates XML for a menu by wrapping tags around all of the menu items properties.	"XML Menu Function" on page 494		
Metadata Functions	Metadata Functions			
Metadata	Displays the metadata for the current Web page.	"Metadata Function" on page 495		
Search Functions				
Search	Inserts a text box with a search button.	"Search Function" on page 497		
Search Display	Creates an area to display search results.	"Search Display Function" on page 499		
Summary Functions				
List Summary	Displays a list of content block summaries.	"List Summary Function" on page 499		
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 500		
RSS List Summary	Creates an RSS feed out of the items in a list summary.	"RSS List Summary Function" on page 501		

Function	Description	More Information
Single Summary	Displays the summary for a content block.	"Single Summary Function" on page 501

## **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 484

This function is basically the same as the Calendar function in ASP. See "Calendar Function" on page 399 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.

<%= ecmEvtCalendar(24, "MonthUpWithEvent") %>

### **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 396 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.

<%= ecmEvtCalendar(24) %>	

## **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 374 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.

<%= ecmCollection(24, "ecmTeaser") %>	

## 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.

<%= ecmCollectionXML (int id, boolean GetHtml) %>

This function is basically the same as the XML Collection function in ASP. See "XML Collection Function" on page 389 for details.

The only difference between the JSP function and the ASP one is the syntax for implementing it. The JSP example is below.

<%= ecmCollectionXML(24, True) %>	

## **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 xsltId) %>

This function is basically the same as the Random Content function in ASP. See "Random Content" on page 394 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 395 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 387 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 329 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:.

## **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 Overrideld) %>

These function is basically the same as the Content Block functions in ASP. See "Content Block Function" on page 319 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**

```
 ecmContentBlock(1) %>
```

### **Dynamic Content Block**

<%= ecmContentBlock(\$id) %>

### XML Content Block

```
</d>
```

## **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	The "id" argument represents the form's ID number, visible when viewing the form from the View Form screen.		
	View Form "Email Contact Form"		
	📝 🕼 🔀 🐂 🍓 🏊 🛍 📽 ← ∣ View: English (U.S.) 🛛 🗸		
	Form Properties Form Post back message		
	Form Title: Email Contact Form		
	FormID: 259		
	Status: Approved (Published)		
	Description:		
formTagReqd	Allows form tags to be added automatically.		
	This is an example of form tags.		
	<form action="#" method="post" name="myform" onsubmit="return&lt;br&gt;EkFmValidate(this);"></form>		
	true = Add form tags automatically		
	false = Add form tags manually		

This function is basically the same as the Form function in ASP. See "Form Function" on page 403 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.

```
<
<%= ecmFormBlock(1, true) %>
```

## 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.

Site Language:	English (US) 🔹
	Danish
	English (US)
	French (Standard)
	German (Standard)

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 415 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	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.

Button	Name	Description
💰 Logout	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	Workarea	When logged in, this button appears under the logout button allowing the user to access their workarea.
Preview	Preview On	Lets the user to preview the entire Web site as if all checked-in content was published.
× Preview	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 317 for details. The only difference is the syntax used to implement the function. This is listed below.

### Login Function Example

```
<%= ecmlogin() %>
```

## 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:

<%= ccmDHTML\_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,"MyMenul" %>

If the user clicks a graphic to invoke the menu, use this syntax:

<%= ecmDHTML\_DropNextGenMenu menu ID number, '<img src=graphic file name>' %>

#### For example:

<%= ecmDHTML\_DropNextGenMenu 1,'<img src=mymenugraphic.jpg>' %>

### **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/CMS400Example/*Workarea/java/. For a description of the ek\_menu.js file, see "ek\_Menu.js" on page 418.

### 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 421 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 407 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>
<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>

</body>

</html>

#### <%= ecmMetaData ("[1][2]","") %>

🖾 metadata[1] - Notepad	_ 🗆 🗙
<u>File E</u> dit F <u>o</u> rmat <u>H</u> elp	
HTML PUBLIC "-//W3C//DTD HTML 4.0<br Transitional//EN"> <html> <head> <title>CMS400 Sample Web Site; CMS400 Login Information</title> <meta content="CMS400; content&lt;br&gt;management; solution; business users; developers;&lt;br&gt;benefits; Ektron; login" name="Keywords"/></head></html>	
 <body> </body> 	

```
<%= ecmMetaData ("[1;title][2;keywords]","") %>
```

```
ile Edit Format Help

ile Concernent He
```

<%= 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 333 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>

## 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 344 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) %>
```

## 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 350 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>
<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 358 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("\\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:

<%= 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 368 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("\\Books\\", 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 364 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></tr>					
<td< td=""><td>&gt;</td><td></td><td></td><td></td><td></td></td<>	>				
	<%=	ecmSingleSummary	("14",true,"	","","")	응>
<td>'D&gt;</td> <td></td> <td></td> <td></td> <td></td>	'D>				

# 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://msdn.microsoft.com/webservices/understanding/ readme/default.aspx

- 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 1.1 SP2, must be installed on the server (for more information, see http://msdn.microsoft.com/ netframework/). • Ektron CMS400.NET, version 4.5 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
webroot/CMS400Example/Workarea/ webservices/ContentWS.asmx	Entry point for Web Service. This Web page displays examples of all methods.
webroot/CMS400Example/bin	Directory for compiled Web Service code
webroot/CMS400Example/bin/WebService.dll	Web Service dll

## **Setup Instructions**

After you install CMS400.NET, you need to open IIS, select CMS400Example/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.

webservice Properties	<u>?</u> ×				
Directory Documents Directory Security HTTP Headers Custo	m Errors				
When connecting to this resource, the content should come from:					
The designated directory					
C A share located on another computer					
C A redirection to a <u>U</u> RL					
Logal Path: \CMS400Sample\webservice					
Script source access     ✓ Log visits     ✓ Read     ✓ Index this resource     ✓ Write     ✓ Directory browsing					
Application Settings					
Application name: CMS400Sample/WebService	Remove				
Starting point: <default td="" we\webservice<=""><td>Configuration</td></default>	Configuration				
Execute Permissions: Scripts only					
Application Protection: Low (IIS Process)	Unjoad				
OK Cancel Appl	Help				

- 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/CMS400Example/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*/ CMS400Example\Workarea\webservices\ContentWS.asmx. For more information, see "The CMS400.NET Content Sample Page" on page 510.

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 513	Search Display information as an HTML string		
GetSearchDisplay	ecmSearchDisplay See Also: "The Search Display Method Parameters" on page 513	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	<ul> <li>Parameters:</li> <li>Root menu id (that is, the gen- erateHTML- Menu value)</li> <li>Menuld - required menuid</li> <li>title - required title</li> </ul>	

GetFormBlock	ecmFormBlock	Form information as an HTML string	<ul> <li>Parameters:</li> <li>formId</li> <li>formdata generated through webservicesUI</li> <li>formtag required flag (set to true to embed <form> tag; otherwise, set to false)</form></li> </ul>
GetEventsByCalendar	ecmEvtCalendar	Calendar information as an HTML string	<ul> <li>Calendar id</li> <li>event id</li> <li>month</li> <li>year</li> <li>url</li> <li>querystring</li> </ul>
Show RandomSummary	ecmShowRandomSum mary	Randomly displays the summary content of a collection item.	collection ID
ShowRandomContent	ecmShowRandomCon tent	Randomly displays the content of a content block in a specified collection	<ul><li>collection ID</li><li>xsltID</li></ul>

AddContent	ecmAddContent	Lets you programmatically add a new content block to Ektron CMS400 without logging in.		Username password Domain content_title content_comment ContentHTML SummaryHTML ContentLan- guage FolderID GoLive EndDate MetaInfo ErrString
			•	ErrString

### The CMS400.NET Content Sample Page

### The CMS400.NET content sample page

(webroot\CMS400Example\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/ CMS400Example/dev.aspx) and clicking Web Services from the bottom of the left frame (illustrated below).





Welcome to the Ektron CMS400.NET Developer Section.

Ektron has compiled a list of several different examples of developing your Web site with Ektron CMS400.NET.

The individual examples, which can be accessed by clicking on their title in the left navigation menu, show how easy it is to create robust Web pages by using "out-of-the-box" functionality including server controls. Other examples show how easy it is to programatically produce Web applications by building on these 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:	
UserID:	
SitePath:	
Preview:	
SiteLanguage:	
	Invoke

#### 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
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.ektron.com/cms/WebService/ContentWS/ecmGetContentBlock"
```

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	Search text used in search method.
Search Type	n/a	Type of search used in search method.  Search All the words All the words Exact Phrase Match Partial Words Options and - all the words or - any of the words exact phrase - exact phrase Note: This parameter is handled automatically by CMS when displayed on Web page. For Web Services, however, this needs to be defined.
Recursive	Search	Defines whether search is recursive <b>1</b> - recursive search <b>0</b> - non-recursive search

Parameter	Where Defined in Application API.xxx	How Implemented by CMS
Starting Folder	Search	Describes path to folder in which search begins. • Here are some examples.
		<ul> <li>\ - root content folder</li> </ul>
		• \Products\RC Cars - begins search in the RC Cars folder.
Allow Fragments	n/a	Selected by user on page that performs search.
Max Number	Secret	Search All the words All the words Exact Phrase Match Partial Words Options O(false) - do not allow fragments. 1 (true) - allow fragments. 1 (true) - allow fragments. If fragments are allowed, and the user enters "AT" as the search string, the search returns topics with these words: pat, ate, hatter. 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. Note: This parameter is handled automatically by CMS when displayed on Web page. For Web Services, this needs to be defined.
Max Number	Search Display	<ul> <li>Specify the maximum number of topics to display when a search is performed. For example:</li> <li>0 - unlimited</li> <li>1 - one result</li> <li>7 - seven results</li> </ul>
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 lpt; margin:2px; width:50%;"
Show Date	Search Display	Specify whether the last modified date is displayed with search results <b>0</b> - date is not displayed. <b>1</b> - 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">
<ErrorString />
<Item>
        <ContentTitle>Plastic Molder #123</ContentTitle>
        <ContentTitle>Plastic Molder #123</ContentTitle>
<?xml version="1.0"?> <jobposting> <job-title>Plastics Molder</job-title>
<job-number>123</job-number> <description>RC International, a leading
supplier of consumer level radio controlled vehicles, is looking for an
```

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.</description> <qualifications> > 3 years experience in commercial grade plastic molding technologies. > Excellent written and oral communication skills. > Ability to work with very little supervision. > An AS degree in chemistry. </qualifications> <offering> First and second shift opportunities exist. RC International offers competitive pay, an extensive benefits package including a 401k plan, profit sharing, and health coverage.

```
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\CMS400Example\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/CMS400Example/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
```

Νοτε

TE 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 506.)

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 of its Sportster. The RC Sportster is designed to compete with the entry and mid level RC racer

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:	
	Invoke

#### 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: 192.168.0.219
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "urn:ektron:webservices:content:cms300:v21/GetContentBlock"
```

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 flagship product, the RC 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 the 3rd annual RC 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 first product, RC Sportster. The RC Sportster is designed to compete with the entry and mid level RC racer currently on the market.

### The News.aspx Sample (XML)

The news.aspx sample file in the /cmd400Example/WebServiceDemo folder includes all elements described in "The News.aspx Sample (non-XML)" on page 516 *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.

```
strHTML = strHTML & "<b>Content Title: </b>" &
    glsCms400ListSummaryResult.Item(iLoop).ContentTitle & "
strHTML = strHTML & "<b>Date Created: </b>" &
    glsCms400ListSummaryResult.Item(iLoop).DisplayDateCreated & "
tr>"
strHTML = strHTML & "<b>Date Modified: </b>" &
     glsCms400ListSummaryResult.Item(iLoop).DisplayDateModified & "
tr>"
strHTML = strHTML & "<b>Modified By: </b>" &
    glsCms400ListSummaryResult.Item(iLoop).EditorFname & " " &
    glsCms400ListSummaryResult.Item(iLoop).EditorLname & "
strHTML = strHTML & "<b>Content Teaser: </b>
    strHTML = strHTML & "" &
glsCms400ListSummaryResult.Item(iLoop).ContentTeaser & "
strHTML = strHTML & " "
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 flagship product, the RC

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:

Cheetah.

RC International, the fastest growing company in the RC industry, announces plans to host the 3rd annual RC conference in Atlanta, Georgia.

Content ID: 9 Content Title: RC International Ships RC Sportster Date Created: 25-Apr-2002 04:42:10 PM Date Modified: 27-Auq-2002 02:46:12 PM

The lower portion of the page also displays the content block in a ListSummary format, as described in"Getting Content Blocks" on page 517.

### 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 flagship product, the RC 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 the 3rd annual RC 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 first product, RC Sportster. The RC Sportster is designed to compete with the entry and mid level RC racer currently on the market.

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 = "<b>Content Block Information from GetContentBlock
Web Service: </b>"
strHTML = strHTML & "<b>Content Block Title: </b>" &
gcbCms400ContentXml.Item.ContentTitle & "<//r>
strHTML = strHTML & "<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 (\text{Len}(\text{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.
```

Latest News

Content Block Information from GetContentBlock Web Service: Content Block Title: RC International Announces the RC Cheetah Xslt Used for Rendering: None Used (HTML COntent assumed)

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 & "<b>Raw HTML (or XML) from GetContentBlock Web
Service: </b>"
strHTML = strHTML & "& "& strHTML = strHTML & "& strRawHtml & ""
strHTML = strHTML & "& strRawHtml & "
```

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 & "<b>Processed HTML (or XML) from ContentBlock
Web Service: </b>
    strHTML = strHTML & "" & strProcessedHtml & "
    strHTML = strHTML & ""
```

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 availability of its new flagship product, the RC Cheetah. After nearly two years of development the RC Cheetah is designed to compete at the highest level in the RC racing arena. The excitement level of the hard core enthusiasts can be felt at all the demonstrations of the RC Cheetah.

"The RC Cheetah price/performance ration is simply incredible", stated Bob Dickerson, RC International VP of engineering. "The RC Cheetah has been devastating its competition in all the pre-release races and we expect that to continue...".

The major industry magazines are agreeing with Mr. Dickerson's assessment. RC Magazine has tested the RC Cheetah head-to-head with the competition's best and they concluded:

"There is no comparison between the RC Cheetah and its nearest competitor. In speed, handling, and price the RC Cheetah out shines them all".

The RC Cheetah is now available at all RC International distributors and major Hobby stores. Please contact RC International to locate a local dealer.

### 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)
- Decide what information you want to provide about those blocks, and how to present it. Review"CMS400.NET Methods" on page 506 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 CMS400Example 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 .NET), 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/CMS400Example/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/ CMS400Example 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 <code>yourwebsite/WebServices/</code> 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 .NET

This section introduces some background information about using Microsoft's Visual Studio .NET. For more information, use the help feature installed with Visual Studio.NET 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:DataGrid id="DataGrid1" style="Z-INDEX: 103; LEFT: 208px; POSITION:
  absolute; TOP: 152px" runat="server"></asp:DataGrid>
  </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 DataGrid, 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.

# **Additional Information**

# **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

# 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

Ektron, Inc. 5 Northern Blvd., Suite 6 Amherst, NH 03031 Email: sales@ektron.com Tel: +1 603-594-0249 Fax: +1 603-594-0258