

Index

COPYRIGHTED MATERIAL

Index

A

`abort()` **method (XMLHttpRequest object), 92**

`abs(num)` **function (XSLT), 370**

activating Ajax.NET Pro library, 163

Active Server Pages (ASP), 3

Adaptive Path, 8

adding

classes, 274–275

namespaces, 274–275

projects to solutions, 180–181

`adjust-dateTime-to-timezone`
(`datetime`, `timezone`) **function (XSLT), 369**

`adjust-date-to-timezone`(`datetime`,
`timezone`) **function (XSLT), 369**

`adjust-time-to-timezone`(`datetime`,
`timezone`) **function (XSLT), 369**

Ajax

advantages of, 6–7

Atlas support, 249, 251

browser support, 8

defined, 1, 6

development history, 7

DHTML/DOM support, 8

JavaScript, 8

JavaScript Object Notation (JSON), 8

problems it won't solve, 9

processing, 6

UI (user interface), 7

user context, 7

user experience, 5

XML, 8, 122–123

XMLHttpRequest object, 8

AjaxHandlerFactory, **191**

Ajax.NET Pro converters, 193–194

Ajax.NET Pro library

activating, 163

Ajax/AjaxSettingsSectionHandler
.cs, 188–191

AjaxHandlerFactory, 191

AjaxPro.AjaxMethod() attribute,
163–164, 191–192

Compiled Library, 160

downloading, 180

HTTP page handlers, 182–183

JavaScript, 192

Library Source Code, 160, 180

preparing applications to use it, 160–161,
174–175, 180–182

references, 160–161

registering page class, 163, 183–186

using in applications, 161–164

Web.Config settings, 182, 186–188

AjaxPro.AjaxMethod() attribute

AjaxPro.AjaxMethod() **attribute, 163–164, 191–192**

alert() **function (JavaScript), 13–14**

alert **statements, 333**

Anthem.NET server-side framework

client functions, 237–239

complex controls, 234–235

custom attributes, 235–237

Hello World example, 232–233

invoke page method, 239–242

micro-content, 242–244

server exceptions, 244–246

setup, 231–232

appending nodes, 63

apply-imports **element (XSLT), 363**

apply-templates **element (XSLT), 363**

April CTP (Atlas project), 249

architectural distinctions in server-side frameworks, 203–207

arguments (JavaScript functions), 34

Array **class, 267**

arrays in JavaScript, 38

ASP (Active Server Pages), 3

ASP.NET

browser compatibility, 132

callbacks

browser support, 132–133

ClientScript object, 134–135

defined, 127

DetailsView control, 132

enabling pages for callbacks, 145–146

error handling, 141–144

examples, 140–141, 150–154

GridView control, 129–131

ICallbackContainer interface,
133, 154–156

ICallbackEventHandler interface,
133–137

initiating the asynchronous process from the
browser, 138

limitations on returning complex data in XML,
154

obtaining a callback reference, 135–136
receiving and processing the results,
138–150

TreeView control, 128–129

controls

DetailsView, 132

GridView, 129–131

TreeView, 128–129

debugging

breakpoints, 331–332

ease of, 329–330

Visual Studio .NET 2005, 330–332

development environment, 2–5

Easy Search component, 9

integration with Atlas

authentication, 311–313, 315–319

authorization, 311, 314, 319–321

Drag and Drop implementation, 328

membership service, 314

profiles, 311, 314, 322–328

roles, 311, 314, 319–321

Web Parts, 314

interfaces

ICallbackContainer, 133, 154–156

ICallbackEventHandler, 133–137

XML, 154

ASP.NET Forums, 283

Asynchronous Client Script Callbacks.

See callbacks

asynchronous execution, 170–172

Asynchronous JavaScript And XML. See Ajax

asynchronous operations

Atlas project, 251

XMLHttpRequest object, 82–84, 88–92

Atlas project

Ajax support, 249, 251

April CTP, 249

asynchronous operations, 251

blogs, 283

classes

adding, 274–275

Array, 267

- Boolean, 268
- Date, 267
- inheritance, 276–277
- Number, 268
- Object, 267
- RegExp, 268
- registering, 273–274
- String, 268
- System.Data.DataColumn, 269–271
- System.Data.DataRow, 272
- System.Data.DataTable, 270–272
- System.Net.WebRequest, 268–269
- Web.Enum, 280
- client-side components, 250
- client-side scripts, 283
- control extenders
 - AutoComplete, 294–297
 - Drag and Drop, 328
 - functionality, 294
- controls
 - button control, 285–286
 - custom validators, 306–307
 - customValidator control, 303
 - functionality, 285
 - listView control, 287
 - rangeValidator control, 303, 305
 - regexValidator control, 303, 305–306
 - requiredFieldValidator control, 303–305
 - ScriptManager control, 252–253
 - server controls, 288–292
 - System.UI.Data namespace, 287
 - timer control, 292–294
 - typeValidator control, 303, 305
 - UpdatePanel control, 288–292
 - validationErrorLabel control, 303
 - validationGroup control, 308–310
- data binding
 - declarative, 297–299
 - defined, 297
 - directions, 302
 - programmatic, 299–302
 - System.BindingDirection enumeration, 302
 - transformations, 302
 - validation of data input, 303–310
- datatypes, 257–262
- debugging, 281–283
- documentation, 273
- enumerations, 280–281, 302
- integration with ASP.NET services
 - authentication, 311–313, 315–319
 - authorization, 311, 314, 319–321
 - Drag and Drop implementation, 328
 - membership service, 314
 - profiles, 311, 314, 322–328
 - roles, 311, 314, 319–321
 - Web Parts, 314
- interfaces, 277–280
- JavaScript extensions, 267–268
- JavaScript files, 250–251
- March CTP, 249
- Microsoft.Web.Atlas.dll server-side assembly, 250
- namespaces
 - adding, 274–275
 - registering, 273
 - System.Data, 269
 - System.Net, 268
- passing parameters, 257–258
- resources, 283
- server-side support, 250
- synchronous operations, 251–252
- UTC encoding for dates and time, 264
- web services
 - caching, 263–265
 - communicating with, 253–257
 - exposing from a web form, 265–266
 - WebResource.axd HTTP handler, 250
- attacks on security, 18**
- attribute **element (XSLT), 363**
- attributes**
 - AjaxPro.AjaxMethod(), 163–164, 191–192
 - display, 24

attributes (continued)

- id, 21
- language, 32
- Serializable(), 169–170, 194
- size, 21
- src, 20–21, 32
- type, 32
- XML, 105

attribute-set **element (XSLT), 363**

authentication, 311–313, 315–319

authorization, 311, 314, 319–321

AutoComplete **extender control, 294–297**

B

base-uri() **function (XSLT), 366**

basic authentication, 312–313

Bauwer, Wilco (intern and blogger on Atlas project team), 283

Beginning XML, Third Edition (Wiley), 103

binding data

- declarative, 297–299
- defined, 297
- directions, 302
- programmatic, 299–302
- Sys.BindingDirection enumeration, 302
- transformations, 302
- validation of data input, 303–310

blogs about Microsoft Atlas project, 283

Boolean **class, 268**

Boolean **datatype, 168**

boolean(blarg) **function (XSLT), 367**

break **statement (JavaScript), 42–43**

breaking loops, 42–43

breakpoints, 331–332, 336–337

Browser **object**

- SupportsCallback property, 132–133
- SupportsXmlHttpRequest property, 132–133

Browser Object Model, 51–53

browsers

- Ajax support, 8
- ASP.NET, 132
- callback support, 132–133
- checking type, 79–80
- debugging tools, 347–351
- Document Object Model (DOM), 30–31
- JavaScript support, 30–31, 33
- object model standardization, 53
- XMLHttpRequest object support, 78–80

BuildHtmlTable() **function (JavaScript), 169**

BuildMultiplicationTable() **function (JavaScript), 169**

button control, 285–286

C

caching web services, 263–264

callbacks

- browser support, 132–133
- ClientScript object, 134–135
- defined, 127, 170–172
- DetailsView control, 132
- enabling pages for callbacks, 145–146
- error handling, 141–144
- examples, 140–141, 150–154
- GridView control, 129–131
- ICallbackContainer interface, 133, 154–156
- ICallbackEventHandler interface, 133–137
- initiating the asynchronous process from the browser, 138
- limitations on returning complex data in XML, 154
- obtaining a callback reference, 135–136
- receiving and processing the results
 - complex data, 144–150
 - simple data, 138–144
- TreeView control, 128

calling functions in JavaScript, 33

call-template element (XSLT), 363

camel case naming convention, 70**Cascading Style Sheets (CSS)**

DHTML (Dynamic HTML), 11

DOM (Document Object Model), 70–74

case keyword (JavaScript), 41

case-sensitivity

JavaScript, 13, 35

XHTML, 35

ceiling(num) function (XSLT), 370

ChangeImage() method, 162

ChangeMe() function (JavaScript), 20–21,
165–166, 170–171**character references in XML, 108****checking browser type, 79–80**

choose element (XSLT), 115–116

classes

adding, 274–275

Array, 267

Boolean, 268

custom classes, 169–170

custom converter classes, 194

Date, 267

inheritance, 276–277

Number, 268

Object, 267

proxy classes

defined, 166

generating, 175–177, 191–192

RegExp, 268

registering, 273–274

String, 268

Sys.Bindings(), 302

Sys.Data.DataColumn, 269–271

Sys.Data.DataRow, 272

Sys.Data.DataTable, 270–272

Sys.Net.WebRequest, 268–269

Web.Enum, 280

client functions (Anthem.NET server-side framework), 237–239

ClientScript object (ASP.NET), 134–135

client-server development, 1**client-side frameworks**

defined, 195–196

HTMLHttpRequest, 199–201

MochiKit, 201–203

Sarissa, 196–199

client-side JavaScript debugging

Alert statements, 333

breakpoints, 336–337

Command window, 346–347

debugger keyword, 342–344

difficulties of, 332

stepping through code, 340–342

Visual Studio .NET, 334–347

Watch window, 344–346

client-side scripts in Atlas, 283**cloning nodes, 63–64****code**

Ajax.NET Pro library, 160, 180

JavaScript, 35–36

code blocks in JavaScript, 12, 34**code samples**

downloading, 13, 29, 77, 103

port numbers, 78

codepoint-equal(item1, item2)
function (XSLT), 371codepoints-to-string(int1,
int2, ...) function (XSLT), 371**COM (Component Object Model), 3****ComfortASP.NET server-side framework**

complex controls, 211–213

downloading, 208

Hello World example, 210–211

Manager control, 213–216

PanelUpdater control, 216–219

setup, 208–209

Command window, 346–347

comment **element (XSLT), 363**

compare(item1, item2) **function (XSLT), 371**

compare(item1, item2, collation)
function (XSLT), 371

Compiled Library (Ajax.NET Pro library), 160

compiled programs, 11–12

complex controls

Anthem.NET server-side framework, 234–235

ComfortASP.NET server-side framework,
211–213

Component Object Model (COM), 3

composite types in JavaScript, 37–38

concat(string1, string2, ...)

function (XSLT), 371

confirm() **function (JavaScript), 17–18**

constructor **property (JavaScript objects), 44**

constructors (JavaScript), 47–48

contains(string1, string2) **function (XSLT), 372**

context menus, 26–27

context **property (response object), 164–165, 171–172**

continue **statement (JavaScript), 43**

continuous loops, 43

control extenders (Atlas)

AutoComplete, 294–297

Drag and Drop, 328

functionality, 294

controls (ASP.NET)

DetailsView, 132

GridView, 129–131

TreeView, 128–129

controls (Atlas)

button control, 285–286

custom validators, 306–307

customValidator control, 303

functionality, 285

listView control, 287

rangeValidator control, 303, 305

regexValidator control, 303, 305–306

requiredFieldValidator control,
303–305

ScriptManager control, 252–253

server controls, 288–292

Sys.UI.Data namespace, 287

timer control, 292–294

typeValidator control, 303, 305

UpdatePanel control, 288–292

validationErrorLabel control, 303

validationGroup control, 308–310

controls (ComfortASP.NET server-side framework)

complex, 211–213

Manager, 213–216

PanelUpdater control, 216–219

converting objects between .NET and JavaScript Object Notation (JSON), 193–194

copy **element (XSLT), 364**

copying nodes, 63

copy-of **element (XSLT), 364**

CreateCustomObject() **function (JavaScript), 43**

CreateXmlHttpRequestObject()
function (JavaScript), 80

creating nodes, 62

Cross Site Scripting (XSS), 18

CSS (Cascading Style Sheets)

DHTML (Dynamic HTML), 11

DOM (Document Object Model), 70–74

current **function (XSLT), 366**

current-date() **function (XSLT), 367**

current-dateTime() **function (XSLT), 367**

current-time() **function (XSLT), 367**

custom attributes (Anthem.NET server-side framework), 235–237

custom classes, 169–170

custom converter classes, 194

custom HTTP handlers, 94

custom objects, 169–170

custom validators, 306–307

customValidator **control, 303**

D**data binding**

- declarative, 297–299
- defined, 297
- directions, 302
- programmatic, 299–302
- `Sys.BindingDirection` enumeration, 302
- transformations, 302
- validation of data input, 303–310

`data(item, ...)` **function (XSLT), 366**

`DataSet` **datatype, 168**

datasets, 258–260

`DataTable` **datatype, 168–169**

datatypes

- Ajax.NET Pro, 167–169
- Atlas, 257–262
- JavaScript, 36

`Date` **class, 267**

`DateTime` **datatype, 168**

`dateTime(date, time)` **function (XSLT), 367**

`day-from-date(date)` **function (XSLT), 368**

`day-from-dateTime(datetime)` **function (XSLT), 368**

`debug.dump` **command, 281–282**

`debugger` **keyword (JavaScript), 342–344**

debugging

- ASP.NET (server-side)
 - breakpoints, 331–332
 - ease of, 329–330
 - Visual Studio .NET 2005, 330–332
- Atlas, 281–283
- defined, 329
- Fiddler network traffic monitor, 353–361
- JavaScript (client-side)
 - Alert statements, 333
 - breakpoints, 336–337
 - Command window, 346–347
 - debugger keyword, 342–344
 - difficulties of, 332
 - stepping through code, 340–342
 - Visual Studio .NET, 334–347
 - Watch window, 344–346

- network traffic, 351–353

- web browser tools, 347–351

`decimal-format` **element (XSLT), 364**

declarative data binding, 297–299

`default-collation()` **function (XSLT), 367**

defensive programming, 48

`delete` **keyword (JavaScript), 38, 43–44**

`DetailsView` **control (ASP.NET), 132**

development history

- Ajax, 7
- JavaScript, 30
- XML, 104–105
- `XMLHttpRequest` object, 78

development trends

- Ajax, 1, 5–7
- ASP.NET, 2–5
- client-server development, 1
- intelligent client-server development, 1
- terminal-style development, 1–2

DHTML (Dynamic HTML)

- CSS (Cascading Style Sheets), 11
- defined, 11
- HTML (Hypertext Markup Language), 11
- JavaScript, 11
- support, 8

direct scripting

- Anthem.NET server-side framework, 242–244
- MagicAjax server-side framework, 226–231

directions for data binding, 302

`display` **attribute, 24**

`distinct-values((item1, item2, ...), collation)` **function (XSLT), 370**

`div` **element, 23–24**

`document` **function (XSLT), 366**

Document Object Model (DOM). See DOM (Document Object Model)

document tree, 57–62

Document Type Definition (DTD), 108–109
documentation for Microsoft Atlas project, 273

`document.getElementById()` **function (JavaScript), 21–22**

`document-uri(node)` **function (XSLT), 366**

`document.write()` **function (JavaScript), 14**

DOM (Document Object Model)

- browsers, 30–31
- camel case naming convention, 70
- CSS (Cascading Style Sheets), 70–74
- defined, 51–53
- document tree, 57–62
- levels, 53–57
- nodes
 - appending, 63
 - cloning, 63–64
 - copying, 63
 - creating, 62
 - inserting, 63
 - properties, 64–69
 - removing, 63
 - replacing, 63
- object collection support, 69
- parsing XML, 111
- standardization, 31

Double **datatype, 168**

double quotation marks in JavaScript, 16

`do/while` **loop (JavaScript), 41–42**

downloading

- Ajax.NET Pro library, 180
- code samples, 13, 29, 77, 103
- ComfortASP.NET server-side framework, 208
- Venkman (Firefox debugging tool), 350–351

Drag and Drop extender control, 328

drop-down lists, 21–22

DTD (Document Type Definition), 108–109

Dynamic HTML (DHTML)

- CSS (Cascading Style Sheets), 11
- defined, 11
- Hypertext Markup Language (HTML), 11
- JavaScript, 11
- support, 8

dynamic typing in JavaScript, 37

E

Easy Search component (ASP.NET), 9

ECMA (European Computer Manufacturers Association), 30

ECMAScript, 30

Eich, Brendan (creator of JavaScript), 30

`element` **element (XSLT), 364**

`element-available` **function (XSLT), 366**

elements (HTML)

- `div`, 23–24
- `script`, 13–14, 32–33
- `select`, 21–22
- `span`, 23–24

elements (XML), 105

elements (XSLT)

- `apply-imports`, 363
- `apply-templates`, 363
- `attribute`, 363
- `attribute-set`, 363
- `call-template`, 363
- `choose`, 115–116
- `comment`, 363
- `copy`, 364
- `copy-of`, 364
- `decimal-format`, 364
- `element`, 364
- `fallback`, 364
- `for-each`, 115
- `if`, 115
- `import`, 364
- `include`, 364
- `key`, 364
- `message`, 364
- `namespace-alias`, 364
- `number`, 364
- `otherwise`, 364
- `output`, 364
- `param`, 364
- `preserve-space`, 365

processing-instruction, 365
 sort, 115
 strip-space, 365
 stylesheet, 365
 template, 114–115
 text, 365
 transform, 365
 value-of, 115
 variable, 365
 when, 365
 with-param, 365
empty(item1, item2, ...) function (XSLT), 370
enabling pages for callbacks, 145–146
encapsulation, 46
endless loops, 15
end-with(string1, string2) function (XSLT), 372
entity references in XML, 108
enumerations, 280–281, 302
error() function (XSLT), 369
error handling for callbacks, 141–142, 144
error property (response object), 164–165, 173–174
error(error, description, error-object) function (XSLT), 369
error(error, description) function (XSLT), 369
error(error) function (XSLT), 369
escape codes, 16, 36–37
escape-uri(stringurl, escapes) function (XSLT), 372
European Computer Manufacturers Association (ECMA), 30
event handlers
 defined, 34–35
 onclick, 18–19
 onload, 16–17
 onsubmit, 19
 preventing events from bubbling up, 19

events
 onclick, 34–35
 onError, 172
 onLoading, 172–173
 onStateChanged, 172
 onTimeout, 172
exists(item1, item2, ...) function (XSLT), 370
exposing web services from a web form, 265–266
expressions, 39–40
extend property (response object), 164–165
extender controls (Atlas)
 AutoComplete, 294–297
 Drag and Drop, 328
 functionality, 294
Extensible Application Markup Language (XAML), 5
Extensible Markup Language. See XML
Extensible Stylesheet Language Transformations. See XSLT
extensions (XSLT), 120

F

fallback element (XSLT), 364
false() function (XSLT), 367
Fiddler network traffic monitor, 353–361
Firefox
 DOM Inspector tool, 349–350
 Venkman debugging tool, 350–351
Flash, 5
floor(num) function (XSLT), 370
flow control statements (JavaScript), 40–41
for loop (JavaScript), 14–15, 41–42, 283
for-each element (XSLT), 115
for/in loop (JavaScript), 42
format-number function (XSLT), 366
forms-based authentication, 312–313

frameworks

- Ajax.NET Pro
 - activating, 163
 - Ajax/AjaxSettingsSectionHandler.cs, 188–191
 - AjaxHandlerFactory, 191
 - AjaxPro.AjaxMethod() attribute, 163–164, 191–192
 - Compiled Library, 160
 - downloading, 180
 - HTTP page handlers, 182–183
 - JavaScript, 192
 - Library Source Code, 160, 180
 - preparing applications to use it, 160–161, 174–175, 180–182
 - references, 160–161
 - registering page class, 163, 183–186
 - using in applications, 161–164
 - Web.Config settings, 182, 186–188
- client-side
 - defined, 195–196
 - HTMLHttpRequest, 199–201
 - MochiKit, 201–203
 - Sarissa, 196–199
- server-side frameworks
 - Anthem.NET, 231–246
 - architectural distinctions, 203–207
 - ComfortASP.NET, 208–219
 - MagicAjax, 219–231
- FrontPage (Microsoft), 3**
- function keyword (JavaScript), 45
- function literals (JavaScript), 45–46**
- function-available function (XSLT), 366
- functions (JavaScript)**
 - alert(), 13–14
 - arguments, 34
 - BuildHtmlTable(), 169
 - BuildMultiplicationTable(), 169
 - calling, 33
 - ChangeImage(), 162
 - ChangeMe(), 20–21, 165–166, 170–171
 - confirm(), 17–18
 - CreateCustomObject(), 43
 - CreateXmlHttpRequestObject(), 80
 - defined, 20–21
 - document.getElementById(), 21–22
 - document.write(), 14
 - HandleClick(), 35
 - hasOwnProperty(nameOfProperty), 45
 - isPrototypeOf(obj), 45
 - OnServerCallError, 142
 - parameters, 34
 - prompt(), 17
 - propertyIsEnumerable(nameOfProperty), 45
 - RegisterCommonAjax(), 184–186
 - RegisterTypeForAjax(), 183–184
 - RenderControl(), 167
 - return values, 34
 - ShowHide(), 25–26
 - toLocaleString(), 44
 - toString(), 44
 - Type.registerNameSpace(), 273
 - UpdateText(), 25–26
 - valueOf(), 44
- functions (XML DOM)**
 - selectNodes(), 87–88, 91–92
 - selectSingleNode(), 87–88
- functions (XMLHttpRequest object)**
 - abort(), 92
 - getAllResponseHeaders(), 92
 - getResponseHeader("headername"), 92
 - open("method", "URL", "async", "uname", "pswd"), 92
 - send(content), 92
 - setRequestHeader("label", "value"), 92
- functions (XSLT)**
 - abs(num), 370
 - adjust-date-time-to-timezone(date time, timezone), 369
 - adjust-date-to-timezone(datetime, timezone), 369

- adjust-time-to-timezone(datetime, timezone), 369
- base-uri(), 366
- boolean(blarg), 367
- ceiling(num), 370
- codepoint-equal(item1, item2), 371
- codepoints-to-string(int1, int2, ...), 371
- compare(item1, item2), 371
- compare(item1, item2, collation), 371
- concat(string1, string2, ...), 371
- contains(string1, string2), 372
- current, 366
- current-date(), 367
- current-dateTime(), 367
- current-time(), 367
- data(item, ...), 366
- dateTime(date, time), 367
- day-from-date(date), 368
- day-from-dateTime(datetime), 368
- default-collation(), 367
- distinct-values((item1, item2, ...), collation), 370
- document, 366
- document-uri(node), 366
- element-available, 366
- empty(item1, item2, ...), 370
- end-with(string1, string2), 372
- error(), 369
- error(error), 369
- error(error, description), 369
- error(error, description, error-object), 369
- escape-uri(stringurl, escres), 372
- exists(item1, item2, ...), 370
- false(), 367
- floor(num), 370
- format-number, 366
- function-available, 366
- generate-id, 366
- hours-from-dateTime(datetime), 368
- hours-from-duration(datetime duration), 368
- hours-from-time(time), 368
- implicit-timezone(), 367
- index-of((item1, item2, ...), searchitem), 370
- insert-before((item1, item2, ...), position, inserts), 370
- key, 366
- lang(lang), 369
- last(), 367
- local-name(), 369
- local-name(nodeset), 369
- lower-case(string), 372
- matches(string, pattern), 372
- minutes-from-dateTime(datetime), 368
- minutes-from-duration(datetime duration), 368
- minutes-from-time(time), 368
- month-from-date(date), 368
- month-from-dateTime(datetime), 368
- months-from-duration(datetime duration), 368
- name(), 369
- name(nodeset), 369
- namespace-uri(), 369
- namespace-uri(nodeset), 369
- nilled(node), 366
- node-name(node), 366
- node-set, 366
- normalize-space(), 371
- normalize-space(string), 371
- normalize-unicode(), 372
- not(blarg), 367
- number(arg), 370
- position(), 367
- remove((item1, item2, ...), position), 370
- replace(string, pattern, replacement), 372
- resolve-uri(relative, base), 372

functions (XSLT) (continued)

reverse((item1, item2, ...)), 370
root(), 369
root(node), 369
round-half-to-even(num), 370
round(num), 370
seconds-from-dateTime(datetime), 368
seconds-from-duration(datetime duration), 368
seconds-from-time(time), 368
starts-with(string1, string2), 372
static-base-uri(), 367
string(argument), 371
string-join((string1, string2, ...), separator), 371
string-length(), 371
string-length(string), 371
string-to-codepoints(string), 371
subsequence((item1, item2, ...), start, length), 371
substring-after(string1, string2), 372
substring-before(string1, string2), 372
substring(string, start), 371
substring(string, start, length), 371
system-property, 366
timezone-from-date(date), 368
timezone-from-time(time), 368
tokenize(string, patternstring), 372
trace(value, label), 369
translate(string1, string2, string3), 372
true(), 367
unordered((item1, item2, ...)), 371
unparsed-entry-uri, 366
upper-case(string), 372
year-from-date(date), 368

year-from-dateTime(datetime), 368
years-from-duration(datetime duration), 367

G

Garret, Jesse James, Adaptive Path, 8
generate-id **function (XSLT), 366**
generating proxy classes, 175–177, 191–192
getAllResponseHeaders() **method (XMLHttpRequest object), 92**
getElementById() **function (JavaScript), 21–22**
getResponseHeader("headername") **method (XMLHttpRequest object), 92**
Google
GMail, 9, 78
Google Maps, 7, 9, 78
Google Suggest, 7, 9
GridView **control (ASP.NET), 129–131**
group validation, 308–310
Guthrie, Scott (blogger), 283

H

HandleClick() **function (JavaScript), 35**
hasOwnProperty(nameOfProperty) **function (JavaScript), 45**
Hello World example
Anthem.NET server-side framework, 232–233
ComfortASP.NET server-side framework, 210–211
MagicAjax server-side framework, 221–222
Hotmail Kahuna update (Microsoft), 9
hours-from-dateTime(datetime) **function (XSLT), 368**
hours-from-duration(datetime duration) **function (XSLT), 368**
hours-from-time(time) **function (XSLT), 368**

HTML (Hypertext Markup Language)

DHTML (Dynamic HTML), 11
 JavaScript, 14

HTML injection attacks, 18**HTMLHttpRequest library, 199–201****HTTP handlers**

custom, 94
 WebResource.axd HTTP handler, 250
 XMLHttpRequest object, 94–97

HTTP page handlers, Ajax.NET Pro library, 182–183**HTTP status codes, 91****HTTP traffic, 351–353****HTTP web services, 103****Hypertext Markup Language (HTML)**

Dynamic HTML (DHTML), 11
 JavaScript, 14

I

ICallbackContainer **interface, 133, 154–156**

ICallbackEventHandler **interface, 133–137**

id **attribute, 21**

IDC (Internet Database Connector), 3

if **element (XSLT), 115**

if-else **statement (JavaScript), 40–41**

images, 19–20

implicit-timezone() **function (XSLT), 367**

import **element (XSLT), 364**

include **element (XSLT), 364**

index-of((item1, item2, ...),
 searchitem) **function (XSLT), 370**

inheritance, 50–51, 276–277

**initiating the asynchronous callback process
 from the browser, 138**

innerHTML **property, 23–26**

insert-before((item1, item2, ...),
 position, inserts) **function (XSLT),
 370**

inserting nodes, 63

integers, 168

intelligent client-server development, 1

interfaces

ASP.NET

ICallbackContainer, 133, 154–156

ICallbackEventHandler, 133–137

Atlas, 277–280

**International Organization for Standardization
 (ISO), 30**

Internet Database Connector (IDC), 3

Internet Explorer Developer Toolbar, 348–349

interpreted programs, 11–12

Invert **transform, 302**

invisible postback, 88–91

**invoke page method (Anthem.NET server-side
 framework), 239–242**

**ISO (International Organization for
 Standardization), 30**

isPrototypeOf(obj) **function
 (JavaScript), 45**

J**Java**

JavaScript, 12, 30
 user experience, 5

JavaScript

Ajax, 8

Ajax.NET Pro library, 192

arrays, 38

Atlas

extensions, 267–268

files included, 250–251

browser support, 30–31, 33

case-sensitivity, 13, 35

code, 35–36

code blocks, 12, 34

composite types, 37–38

constructors, 47–48

context menus, 26–27

JavaScript (continued)

- datatypes, 36
- debugging
 - Alert statements, 333
 - breakpoints, 336–337
 - Command window, 346–347
 - debugger keyword, 342–344
 - difficulties of, 332
 - stepping through code, 340–342
 - Visual Studio .NET, 334–347
 - Watch window, 344–346
- defensive programming, 48
- defined, 12
- development history, 30
- DHTML (Dynamic HTML), 11
- drop-down lists, 21–22
- dynamic typing, 37
- ECMAScript, 30
- encapsulation, 46
- escape codes, 16, 36–37
- event handlers
 - defined, 34–35
 - onclick, 18–19
 - onload, 16–17
 - onsubmit, 19
 - preventing events from bubbling up, 19
- expressions, 39–40
- function literals, 45–46
- functions
 - alert(), 13–14
 - arguments, 34
 - BuildHtmlTable(), 169
 - BuildMultiplicationTable(), 169
 - calling, 33
 - ChangeMe(), 20–21, 165–166, 170–171
 - confirm(), 17–18
 - CreateCustomObject(), 43
 - CreateXmlHttpRequestObject(), 80
 - defined, 20–21
 - document.getElementById(), 21–22
 - document.write(), 14
 - HandleClick(), 35
 - hasOwnProperty(nameOfProperty), 45
 - isPrototypeOf(obj), 45
 - OnServerError, 142
 - parameters, 34
 - prompt(), 17
 - propertyIsEnumerable(nameOfProperty), 45
 - RegisterCommonAjax(), 184–186
 - RegisterTypeForAjax(), 183–184
 - RenderControl(), 167
 - return values, 34
 - ShowHide(), 25–26
 - toLocaleString(), 44
 - toString(), 44
 - Type.registerNameSpace(), 273
 - UpdateText(), 25–26
 - valueOf(), 44
- HTML (Hypertext Markup Language), 14
- images, 19–20
- inheritance, 50–51
- Java, 12, 30
- keywords
 - case, 41
 - debugger, 342–344
 - delete, 38, 43–44
 - function, 45
 - new, 38
 - return, 19
 - this, 19
 - var, 35, 37
- loops
 - breaking, 42–43
 - continuous, 43
 - do/while, 41–42
 - endless loops, 15
 - for, 14–15, 41–42, 283
 - for/in, 42
 - while, 41–42
- object literals, 46
- objects, 38–39, 43–44

- operators, 39–40
- properties
 - constructor, 44
 - prototype, 44, 49–50
- prototype chaining, 50–51
- prototypes, 46–49
- quotation marks, 16
- script element, 13–14, 32–33
- standardization, 30
- statements
 - Alert, 333
 - break, 42–43
 - continue, 43
 - flow control statements, 40–41
 - if-else, 40
 - switch, 41
 - syntax, 12
 - writing, 13–14
- syntax, 12, 33–34
- user input, 17–18
- variables, 35
- weak typing, 37

JavaScript Object Notation (JSON)

- Ajax, 8
- converting objects between .NET and JavaScript Object Notation (JSON), 193–194
- defined, 103, 124
- example, 125
- layout, 124–125
- web services, 103
- `.js` filename extension, 14

JScript (Microsoft), 30

JSON (JavaScript Object Notation)

- Ajax, 8
- converting objects between .NET and JavaScript Object Notation (JSON), 193–194
- defined, 103, 124
- example, 125
- layout, 124–125
- web services, 103

K

key **element (XSLT)**, 364

key **function (XSLT)**, 366

keywords (JavaScript)

- case, 41
- debugger, 342–344
- delete, 38, 43–44
- function, 45
- new, 38
- return, 19
- this, 19
- var, 35, 37

Kothari, Nikhil (blogger), 283

L

lang (lang) **function (XSLT)**, 369

language **attribute**, 32

last () **function (XSLT)**, 367

libraries

- Ajax.NET Pro library
 - activating, 163
 - Ajax/AjaxSettingsSection Handler.cs, 188–191
 - AjaxHandlerFactory, 191
 - AjaxPro.AjaxMethod() attribute, 163–164, 191–192
 - Compiled Library, 160
 - downloading, 180
 - HTTP page handlers, 182–183
 - JavaScript, 192
 - Library Source Code, 160, 180
 - preparing applications to use it, 160–161, 174–175, 180–182
 - references, 160–161
 - registering page class, 163, 183–186
 - using in applications, 161–164
 - Web.Config settings, 182, 186–188
- Anthem.NET
 - client functions, 237–239
 - complex controls, 234–235

libraries (continued)

- custom attributes, 235–237
- Hello World example, 232–233
- invoke page method, 239–242
- micro-content, 242–244
- server exceptions, 244–246
- setup, 231–232
- ComfortASP.NET
 - complex controls, 211–212
 - downloading, 208
 - Hello World example, 210–211
 - Manager control, 213–216
 - PanelUpdater control, 216–219
 - setup, 208–209
- HTMLHttpRequest, 199–201
- MagicAjax
 - call helper, 224–226
 - delay message, 222–224
 - Hello World example, 221–222
 - micro-content, 226–231
 - setup, 219–220
- MochiKit, 201–203
- Sarissa, 196–199
- Library Source Code (Ajax.NET Pro library), 160, 180**
- listView **control, 287**
- Live.com (Microsoft), 9**
- local-name() **function (XSLT), 369**
- local-name(nodeset) **function (XSLT), 369**
- loops (JavaScript)**
 - breaking, 42–43
 - continuous, 43
 - do/while, 41–42
 - endless loops, 15
 - for, 14–15, 41–42, 283
 - for/in, 42
 - while, 41–42
- lower-case(string) **function (XSLT), 372**

M

MagicAjax server-side framework

- call helper, 224–226
- delay message, 222–224
- Hello World example, 221–222
- micro-content, 226–231
- setup, 219–220

Manager **control (ComfortASP.NET server-side framework), 213–216**

March CTP (Atlas project), 249

matches(string, pattern) **function (XSLT), 372**

membership service, 314

menus, 26–27

message **element (XSLT), 364**

methods

- overloads, 171
- signature, 171

methods (JavaScript)

- alert(), 13–14
- arguments, 34
- BuildHtmlTable(), 169
- BuildMultiplicationTable(), 169
- calling, 33
- ChangeImage(), 162
- ChangeMe(), 20–21, 165–166, 170–171
- confirm(), 17–18
- CreateCustomObject(), 43
- CreateXmlHttpRequestObject(), 80
- defined, 20–21
- document.getElementById(), 21–22
- document.write(), 14
- HandleClick(), 35
- hasOwnProperty(nameOfProperty), 45
- isPrototypeOf(obj), 45
- OnServerCallError, 142
- parameters, 34
- prompt(), 17

- propertyIsEnumerable(nameOf Property), 45
- RegisterCommonAjax(), 184–186
- RegisterTypeForAjax(), 183–184
- RenderControl(), 167
- return values, 34
- ShowHide(), 25–26
- toLocaleString(), 44
- toString(), 44
- Type.registerNameSpace(), 273
- UpdateText(), 25–26
- valueOf(), 44
- methods (XML DOM)**
 - selectNodes(), 87–88, 91–92
 - selectSingleNode(), 87–88
- methods (XMLHttpRequest object)**
 - abort(), 92
 - getAllResponseHeaders(), 92
 - getResponseHeader("headername"), 92
 - open("method", "URL", "async", "uname", "pswd"), 92
 - send(content), 92
 - setRequestHeader("label", "value"), 92
- methods (XSLT)**
 - abs(num), 370
 - adjust-dateTime-to-timezone(date time, timezone), 369
 - adjust-date-to-timezone(datetime, timezone), 369
 - adjust-time-to-timezone(datetime, timezone), 369
 - base-uri(), 366
 - boolean(blarg), 367
 - ceiling(num), 370
 - codepoint-equal(item1, item2), 371
 - codepoints-to-string(int1, int2, ...), 371
 - compare(item1, item2), 371
 - compare(item1, item2, collation), 371
 - concat(string1, string2, ...), 371
 - contains(string1, string2), 372
 - current, 366
 - current-date(), 367
 - current-dateTime(), 367
 - current-time(), 367
 - data(item, ...), 366
 - dateTime(date, time), 367
 - day-from-date(date), 368
 - day-from-dateTime(datetime), 368
 - default-collation(), 367
 - distinct-values((item1, item2, ...), collation), 370
 - document, 366
 - document-uri(node), 366
 - element-available, 366
 - empty(item1, item2, ...), 370
 - end-with(string1, string2), 372
 - error(), 369
 - error(error), 369
 - error(error, description), 369
 - error(error, description, error-object), 369
 - escape-uri(stringurl, escapes), 372
 - exists(item1, item2, ...), 370
 - false(), 367
 - floor(num), 370
 - format-number, 366
 - function-available, 366
 - generate-id, 366
 - hours-from-dateTime(datetime), 368
 - hours-from-duration(datetime duration), 368
 - hours-from-time(time), 368
 - implicit-timezone(), 367
 - index-of((item1, item2, ...), searchitem), 370
 - insert-before((item1, item2, ...), position, inserts), 370
 - key, 366
 - lang(lang), 369

methods (XSLT) (continued)

`last()`, 367
`local-name()`, 369
`local-name(nodeset)`, 369
`lower-case(string)`, 372
`matches(string, pattern)`, 372
`minutes-from-dateTime(datetime)`, 368
`minutes-from-duration(datetime duration)`, 368
`minutes-from-time(time)`, 368
`month-from-date(date)`, 368
`month-from-dateTime(datetime)`, 368
`months-from-duration(datetime duration)`, 368
`name()`, 369
`name(nodeset)`, 369
`namespace-uri()`, 369
`namespace-uri(nodeset)`, 369
`nilled(node)`, 366
`node-name(node)`, 366
`node-set`, 366
`normalize-space()`, 371
`normalize-space(string)`, 371
`normalize-unicode()`, 372
`not(blarg)`, 367
`number(arg)`, 370
`position()`, 367
`remove((item1, item2, ...), position)`, 370
`replace(string, pattern, replacement)`, 372
`resolve-uri(relative, base)`, 372
`reverse((item1, item2, ...))`, 370
`root()`, 369
`root(node)`, 369
`round-half-to-even(num)`, 370
`round(num)`, 370
`seconds-from-dateTime(datetime)`, 368
`seconds-from-duration(datetime duration)`, 368
`seconds-from-time(time)`, 368
`starts-with(string1, string2)`, 372
`static-base-uri()`, 367
`string(argument)`, 371
`string-join((string1, string2, ...), separator)`, 371
`string-length()`, 371
`string-length(string)`, 371
`string-to-codepoints(string)`, 371
`subsequence((item1, item2, ...), start, length)`, 371
`substring-after(string1, string2)`, 372
`substring-before(string1, string2)`, 372
`substring(string, start)`, 371
`substring(string, start, length)`, 371
`system-property`, 366
`timezone-from-date(date)`, 368
`timezone-from-time(time)`, 368
`tokenize(string, patternstring)`, 372
`trace(value, label)`, 369
`translate(string1, string2, string3)`, 372
`true()`, 367
`unordered((item1, item2, ...))`, 371
`unparsed-entry-uri`, 366
`upper-case(string)`, 372
`year-from-date(date)`, 368
`year-from-dateTime(datetime)`, 368
`years-from-duration(datetime duration)`, 367

micro-content

Anthem.NET server-side framework, 242–244
MagicAjax server-side framework, 226–231

Microsoft

Active Server Pages (ASP), 3
ASP.NET development, 2–5
FrontPage, 3
Hotmail Kahuna update, 9

Internet Database Connector (IDC), 3
 Internet Explorer Developer Toolbar, 348–349
 JScript, 30
 Live.com, 9
 MSN Virtual Earth, 78
 Passport Service, 312–313
 VBScript, 30

Microsoft Atlas project

Ajax support, 249, 251
 April CTP, 249
 asynchronous operations, 251
 blogs, 283
 classes
 adding, 274–275
 Array, 267
 Boolean, 268
 Date, 267
 inheritance, 276–277
 Number, 268
 Object, 267
 RegExp, 268
 registering, 273–274
 String, 268
 Sys.Data.DataColumn, 269–271
 Sys.Data.DataRow, 272
 Sys.Data.DataTable, 270–272
 Sys.Net.WebRequest, 268–269
 Web.Enum, 280
 client-side components, 250
 client-side scripts, 283
 control extenders
 AutoComplete, 294–297
 Drag and Drop, 328
 functionality, 294
 controls
 button control, 285–286
 custom validators, 306–307
 customValidator control, 303
 functionality, 285
 listView control, 287
 rangeValidator control, 303, 305
 regexValidator control, 303, 305–306

requiredFieldValidator control,
 303–305
 ScriptManager control, 252–253
 server controls, 288–292
 Sys.UI.Data namespace, 287
 timer control, 292–294
 typeValidator control, 303, 305
 UpdatePanel control, 288–292
 validationErrorLabel control, 303
 validationGroup control, 308–310
 data binding
 declarative, 297–299
 defined, 297
 directions, 302
 programmatic, 299–302
 Sys.BindingDirection enumeration,
 302
 transformations, 302
 validation of data input, 303–310
 datatypes, 257–262
 debugging, 281–283
 documentation, 273
 enumerations, 280–281, 302
 integration with ASP.NET services
 authentication, 311–313, 315–319
 authorization, 311, 314, 319–321
 Drag and Drop implementation, 328
 membership service, 314
 profiles, 311, 314, 322–328
 roles, 311, 314, 319–321
 Web Parts, 314
 interfaces, 277–280
 JavaScript extensions, 267–268
 JavaScript files, 250–251
 March CTP, 249
 Microsoft.Web.Atlas.dll server-side
 assembly, 250
 namespaces
 adding, 274–275
 registering, 273
 Sys.Data, 269
 Sys.Net, 268

Microsoft Atlas project (continued)

- passing parameters, 257–258
- resources, 283
- server-side support, 250
- synchronous operations, 251–252
- UTC encoding for dates and time, 264
- web services
 - caching, 263–265
 - communicating with, 253–257
 - exposing from a web form, 265–266
- WebResource.axd HTTP handler, 250
- minutes-from-dateTime(datetime) **function (XSLT), 368**
- minutes-from-duration(datetime duration) **function (XSLT), 368**
- minutes-from-time(time) **function (XSLT), 368**
- mismatching parameters, 193**
- MochiKit client-side framework, 201–203**
- month-from-date(date) **function (XSLT), 368**
- month-from-dateTime(datetime) **function (XSLT), 368**
- months-from-duration(datetime duration) **function (XSLT), 368**
- MSN Virtual Earth, 78**

N

- name() **function (XSLT), 369**
- name(nodeset) **function (XSLT), 369**
- namespace-alias **element (XSLT), 364**
- namespaces**
 - adding, 274–275
 - registering, 273
 - System.Data, 269
 - System.Net, 268
 - System.Data, 258
 - System.UI.Data, 287
- namespace-uri() **function (XSLT), 369**
- namespace-uri(nodeset) **function (XSLT), 369**

.NET Framework

- converting objects between .NET and JavaScript Object Notation (JSON), 193–194
- reflection, 176–177
- network traffic, 351–353**
- new **keyword (JavaScript), 38**
- nilled(node) **function (XSLT), 366**
- node-name(node) **function (XSLT), 366**
- nodes**
 - appending, 63
 - cloning, 63–64
 - copying, 63
 - creating, 62
 - inserting, 63
 - properties, 64–69
 - removing, 63
 - replacing, 63
- node-set **function (XSLT), 366**
- normalize-space() **function (XSLT), 371**
- normalize-space(string) **function (XSLT), 371**
- normalize-unicode() **function (XSLT), 372**
- not(blarg) **function (XSLT), 367**
- Notepad, 12**
- Number **class, 268**
- number **element (XSLT), 364**
- number(arg) **function (XSLT), 370**
- numeric character references in XML, 108**
- numeric processing with XSLT, 118–120**

O

- Object **class, 267**
- object literals (JavaScript), 46**
- object model standardization, 31, 53**
- objects**
 - Browser object
 - SupportsCallback property, 132–133
 - SupportsXmlHttp property, 132–133
 - ClientScript object (ASP.NET), 134–135
 - custom objects, 169–170

- JavaScript, 38–39, 43–44
- request object
 - onError event, 172
 - onLoading event, 172–173
 - onStateChanged event, 172
 - onTimeout event, 172
- response object
 - context property, 164–165, 171–172
 - defined, 164
 - error property, 164–165, 173–174
 - extend property, 164–165
 - request property, 164–165, 173
 - value property, 164–165, 167–169
- XML Document Object Model, 87
- XMLHttpRequest object
 - abort() method, 92
 - asynchronous requests, 82–84, 88–92
 - browser support, 78–80
 - development history, 78
 - functionality, 8, 77–78
 - getAllResponseHeaders() method, 92
 - getResponseHeader("headername") method, 92
 - HTTP handlers, 94–97
 - onreadystatechange property, 93
 - open("method", "URL", "async", "uname", "pswd") method, 92
 - passing parameters, 93–94
 - readyState property, 93
 - responseText property, 84–87, 93
 - responseXML property, 84, 87, 93
 - Sarissa client-side framework, 196
 - security, 84
 - send(content) method, 92
 - setRequestHeader("label", "value") method, 92
 - status property, 91–93
 - statusText property, 93
 - synchronous requests, 81–82
 - web services, 97–102
 - XMLHTTP protocol, 78, 80–81
 - obtaining a callback reference, 135–136**
 - onclick **event, 34–35**
 - onclick **event handler (JavaScript), 18–19**
 - onError **event, 172**
 - onload **event handler (JavaScript), 16–17**
 - onLoading **event, 172–173**
 - onreadystatechange **property (XMLHttpRequest object), 93**
 - OnServerError **function (JavaScript), 142**
 - onStateChanged **event, 172**
 - onsubmit **event handler (JavaScript), 19**
 - onTimeout **event, 172**
 - open("method", "URL", "async", "uname", "pswd") **method (XMLHttpRequest object), 92**
 - operators in JavaScript, 39–40**
 - otherwise **element (XSLT), 364**
 - output **element (XSLT), 364**
 - overloads, 171**

P

 - page **class, registering, 163, 183–186**
 - PanelUpdater **control (ComfortASP.NET server-side framework), 216–219**
 - param **element (XSLT), 364**
 - parameters**
 - JavaScript functions, 34
 - mismatching parameters, 193
 - passing
 - Atlas, 257–258
 - XMLHttpRequest object, 93–94
 - parsing XML**
 - Document Object Model (DOM), 111
 - Simple API for XML (SAX), 112
 - passing parameters**
 - Atlas, 257–258
 - XMLHttpRequest object, 93–94
 - Passport authentication, 312–313**
 - Passport Service (Microsoft), 312–313**

port 80, 103**port numbers in code samples, 78****position() function (XSLT), 367****postback delay, 77****preparing applications to use Ajax.NET Pro library, 160–161, 174–175, 180–182****preserve-space element (XSLT), 365****processing**

Ajax, 6

ASP.NET, 5

XSLT, 112–114, 116–120

processing-instruction element (XSLT), 365**profiles, 311, 314, 322–328****programmatic data binding, 299–302****projects, adding to solutions, 180–181****prompt() function (JavaScript), 17****properties**

constructor (JavaScript objects), 44

context (response object), 164–165, 171–172

error (response object), 164–165, 173–174

extend (response object), 164–165

innerHTML, 23–26

nodes, 64–69

onreadystatechange (XMLHttpRequest object), 93

prototype (JavaScript objects), 44, 49–50

readyState (XMLHttpRequest object), 93

request (response object), 164–165, 173

responseText (XMLHttpRequest object), 84–87, 93

responseXML (XMLHttpRequest object), 84, 87, 93

status (XMLHttpRequest object), 91–93

statusText (XMLHttpRequest object), 93

SupportsCallback (Browser object), 132–133

SupportsXmlHttp (Browser object), 132–133

value (response object), 164–165, 167–169

propertyIsEnumerable (nameOf

Property) **function (JavaScript), 45****prototype chaining, 50–51**prototype **property (JavaScript objects), 44, 49–50****prototypes (JavaScript), 46–49****proxy classes**

defined, 166

generating, 175–177, 191–192

Q

quotation marks in JavaScript, 16

R

rangeValidator control, 303, 305**readyState property (XMLHttpRequest object), 93****references to Ajax.NET Pro library, 160–161****reflection, 176–177****RegExp class, 268****regexValidator control, 303, 305–306****RegisterCommonAjax() method, 184–186****registering**

classes, 273–274

namespaces, 273

page class, 163, 183–186

RegisterTypeForAjax() method, 183–184**remove((item1, item2, ...), position) function (XSLT), 370****removing nodes, 63****RenderControl() method, 167****replace(string, pattern, replacement) function (XSLT), 372****replacing nodes, 63****request object events**

onError, 172

onLoading, 172–173

onStateChanged, 172

onTimeout, 172

request **property** (response **object**),
164–165, 173

requiredFieldValidator **control**,
303–305

resolve-uri(relative, base) **function**
(XSLT), 372

response **object**
 defined, 164
 properties
 context, 164–165, 171–172
 error, 164–165, 173–174
 extend, 164–165
 request, 164–165, 173
 value, 164–165, 167–169

responseText **property** (XMLHttpRequest
object), **84–87, 93**

responseXML **property** (XMLHttpRequest
object), **84, 87, 93**

return **keyword** (JavaScript), **19**

return values of JavaScript functions, 34

reverse((item1, item2, ...)) **function**
(XSLT), 370

roles, 311, 314, 319–321

root() **function** (XSLT), **369**

root(node) **function** (XSLT), **369**

round trips
 ASP.NET, 4
 defined, 2

round-half-to-even(num) **function**
(XSLT), 370

round(num) **function** (XSLT), **370**

S

Sarissa client-side framework, 196–199

SAX (Simple API for XML), 112

script **element, 13–14, 32–33**

ScriptManager **control, 252–253**

seconds-from-dateTime(datetime)
function (XSLT), **368**

seconds-from-duration(datetime
 duration) **function** (XSLT), **368**

seconds-from-time(time) **function**
(XSLT), 368

security
 HTML injection attacks, 18
 XMLHttpRequest object, 84

select **element, 21–22**

selectNodes() **function** (XML DOM), **87–88,**
91–92

selectSingleNode() **function** (XML DOM),
87–88

send(content) **method** (XMLHttpRequest
object), **92**

Serializable() **attribute, 169–170, 194**

server controls, 288–292

server-side ASP.NET debugging, 329–332

server-side frameworks
 Anthem.NET
 client functions, 237–239
 complex controls, 234–235
 custom attributes, 235–237
 Hello World example, 232–233
 invoke page method, 239–242
 micro-content, 242–244
 server exceptions, 244–246
 setup, 231–232

architectural distinctions, 203–207

ComfortASP.NET
 complex controls, 211–212
 downloading, 208
 Hello World example, 210–211
 Manager control, 213–216
 PanelUpdater control, 216–219
 setup, 208–209

MagicAjax
 call helper, 224–226
 delay message, 222–224

server-side frameworks (continued)

- Hello World example, 221–222
- micro-content, 226–231
- setup, 219–220

server-side support for Atlas, 250

setRequestHeader("label", "value")
method (XMLHttpRequest object), 92

setting breakpoints, 331–332, 336–337

setup

- Anthem.NET server-side framework, 231–232
- ComfortASP.NET server-side framework, 208–209
- MagicAjax server-side framework, 219–220

ShowHide() **function (JavaScript), 25–26**

signature, 171

Simple API for XML (SAX), 112

Simple Object Access Protocol (SOAP), 101–102

single quotation marks in JavaScript, 16

size **attribute, 21**

solutions, adding to projects, 180–181

sort **element (XSLT), 115**

source code for Ajax.NET Pro library, 160, 180

spaghetti code, 3

span **element, 23–24**

src **attribute, 20–21, 32**

standardization

- JavaScript, 30
- object models, 31, 53

starts-with(string1, string2)
function (XSLT), 372

statements (JavaScript)

- Alert statements, 333
- break, 42–43
- continue, 43
- flow control statements, 40–41
- if-else, 40
- switch, 41
- syntax, 12
- writing, 13–14

static-base-uri() **function (XSLT), 367**

status codes, 91

- status **property (XMLHttpRequest object), 91–93**
- statusText **property (XMLHttpRequest object), 93**

String class, 268

string processing with XSLT, 116–117

string(argument) **function (XSLT), 371**

string-join((string1, string2, ...), separator) **function (XSLT), 371**

string-length() **function (XSLT), 371**

string-length(string) **function (XSLT), 371**

strings, 167–168

string-to-codepoints(string)
function (XSLT), 371

strip-space **element (XSLT), 365**

style sheets. See CSS (Cascading Style Sheets)

stylesheet **element (XSLT), 365**

stylesheet tree (XSLT), 114

subsequence((item1, item2, ...), start, length) **function (XSLT), 371**

substring-after(string1, string2)
function (XSLT), 372

substring-before(string1, string2)
function (XSLT), 372

substring(string, start) **function (XSLT), 371**

substring(string, start, length)
function (XSLT), 371

Suggest (Google), 7, 9

SupportsCallback **property (Browser object), 132–133**

SupportsXmlHttp **property (Browser object), 132–133**

switch **statement (JavaScript), 41**

synchronous execution, 170

synchronous operations

- Atlas project, 251–252
- XMLHttpRequest object, 81–82

syntax of JavaScript, 12, 33–34`Sys.BindingDirection` **enumeration, 302**`Sys.Bindings()` **class, 302**`Sys.Data` **namespace, 269**`Sys.Data.DataColumn` **class, 269–271**`Sys.Data.DataRow` **class, 272**`Sys.Data.DataTable` **class, 270–272**`Sys.Net` **namespace, 268**`Sys.Net.WebRequest` **class, 268–269**`System.Data` **namespace, 258**`system-property` **function (XSLT), 366**`Sys.UI.Data` **namespace, 287****T**`template` **element (XSLT), 114–115****terminal-style development, 1–2**`text` **element (XSLT), 365**`this` **keyword (JavaScript), 19****timer control, 292–294**`timezone-from-date(date)` **function (XSLT), 368**`timezone-from-time(time)` **function (XSLT), 368**`tokenize(string, patternstring)` **function (XSLT), 372**`toLocaleString()` **function (JavaScript), 44**`toString()` **function (JavaScript), 44**`ToSting` **transform, 302**`trace(value, label)` **function (XSLT), 369**`transform` **element (XSLT), 365****transformations in data binding, 302**`translate(string1, string2, string3)` **function (XSLT), 372**`TreeView` **control (ASP.NET), 128–129**`true()` **function (XSLT), 367**`type` **attribute, 32**`Type.registerNameSpace()` **method, 273**`typeValidator` **control, 303, 305****U****UI (user interface), 7**`unordered((item1, item2, ...))` **function (XSLT), 371**`unparsed-entry-uri` **function (XSLT), 366**`UpdatePanel` **control, 288–292**`UpdateText()` **function (JavaScript), 25–26**`upper-case(string)` **function (XSLT), 372****user context**

Ajax, 7

ASP.NET, 5

user experience

Ajax, 5

Flash, 5

invisible postback, 88–91

Java, 5

postback delay, 77

XML-based languages, 5

user input in JavaScript, 17–18**user interface (UI), 7****using Ajax.NET Pro library in applications, 161–164****UTC encoding for dates and time, 264****V****valid HTTP status codes, 91****valid XML documents, 106–108****validating**

data input, 303–310

XML, 110–111

`validationErrorLabel` **control, 303**`validationGroup` **control, 308–310****validators**

built-in, 303–307

custom, 306–307

group, 308–310

`value` **property (response object), 164–165, 167–169**

`value-of` element (XSLT), **115**
`valueOf()` function (JavaScript), **44**
`var` keyword (JavaScript), **35, 37**
variable element (XSLT), **365**
variables in JavaScript, **35**
VBScript (Microsoft), **30**
Venkman (Firefox debugging tool), **350–351**
Virtual Earth (MSN), **78**
Visual Studio .NET
 ASP.NET debugging, **330–332**
 JavaScript debugging, **334–347**

W

Watch window, **344–346**
weak typing in JavaScript, **37**
web browsers
 Ajax support, **8**
 ASP.NET, **132**
 callback support, **132–133**
 checking type, **79–80**
 debugging tools, **347–351**
 Document Object Model (DOM), **30–31**
 JavaScript support, **30–31, 33**
 object model standardization, **53**
 XMLHttpRequest object support, **78–80**
Web Parts, **314**
web services
 Atlas
 caching, **263–265**
 communicating with, **253–257**
 exposing from a web form, **265–266**
 HTTP, **103**
 JSON (JavaScript Object Notation), **103**
 SOAP (Simple Object Access Protocol),
 101–102
 XML, **103**
 XMLHttpRequest object, **97–102**
Web.Config settings for Ajax.NET Pro library,
 182, 186–188

Web.Enum class, **280**
WebResource.axd HTTP handler, **250**
well-formed XML documents, **106–108**
when element (XSLT), **365**
while loop (JavaScript), **41–42**
Windows authentication, **312–313**
with-param element (XSLT), **365**
wrapper library, **159**
writing statements (JavaScript), **13–14**

X

X Path 2.0 Programmer's Reference (Wiley),
 103
X Path Language, **87, 112, 121–122**
XAML (Extensible Application Markup
 Language), **5**
XHTML
 case-sensitivity, **35**
 defined, **32**
XML
 Ajax, **8, 122–123**
 ASP.NET, **154**
 attributes, **105**
 defined, **103**
 development history, **104–105**
 Document Type Definition (DTD), **108–109**
 elements, **105**
 entity references, **108**
 features, **104**
 numeric character references, **108**
 parsing
 Document Object Model (DOM), **111**
 Simple API for XML (SAX), **112**
 validating, **110–111**
 web services, **103**
XML declaration, **105**
XML Document Object Model, **87**
XML documents, **105–108**

XML DOM functions

`selectNodes()`, 87–88, 91–92
`selectSingleNode()`, 87–88

XML Path Language (X Path), 87, 112, 121–122**XML Schema, 106, 109–110****XML Schema Definitions (XSDs), 109–110****XML User Interface Language (XUL), 5****XML-based languages and user experience, 5****XMLHTTP protocol, 78, 80–81****XMLHttpRequest object**

asynchronous requests, 82–84, 88–92
 browser support, 78–80
 development history, 78
 functionality, 8, 77–78
 HTTP handlers, 94–97
 methods
 `abort()`, 92
 `getAllResponseHeaders()`, 92
 `getResponseHeader("headername")`, 92
 `open("method", "URL", "async", "uname", "pswd")`, 92
 `send(content)`, 92
 `setRequestHeader("label", "value")`, 92
 passing parameters, 93–94
 properties
 `onreadystatechange`, 93
 `readyState`, 93
 `responseText`, 84–87, 93
 `responseXML`, 84, 87, 93
 `status`, 91–93
 `statusText`, 93
 Sarissa client-side framework, 196
 security, 84
 synchronous requests, 81–82
 web services, 97–102
 XMLHTTP protocol, 78, 80–81

XSDs (XML Schema Definitions), 109–110**XSLT**

defined, 103, 112
 extensions, 120
 processing, 112–114, 116–120
 stylesheet tree, 114
 X Path Language, 112, 121–122

XSLT 2.0 Programmer's Reference, Third Edition (Wiley), 103**XSLT elements**

`apply-imports`, 363
`apply-templates`, 363
`attribute`, 363
`attribute-set`, 363
`call-template`, 363
`choose`, 115–116
`comment`, 363
`copy`, 364
`copy-of`, 364
`decimal-format`, 364
`element`, 364
`fallback`, 364
`for-each`, 115
`if`, 115
`import`, 364
`include`, 364
`key`, 364
`message`, 364
`namespace-alias`, 364
`number`, 364
`otherwise`, 364
`output`, 364
`param`, 364
`preserve-space`, 365
`processing-instruction`, 365
`sort`, 115
`strip-space`, 365
`stylesheet`, 365
`template`, 114–115
`text`, 365
`transform`, 365

XSLT elements (continued)

XSLT elements (continued)

value-of, 115
variable, 365
when, 365
with-param, 365

XSLT functions

abs(num), 370
adjust-dateTime-to-timezone(date
time, timezone), 369
adjust-date-to-timezone(datetime,
timezone), 369
adjust-time-to-timezone(datetime,
timezone), 369
base-uri(), 366
boolean(blarg), 367
ceiling(num), 370
codepoint-equal(item1, item2), 371
codepoints-to-string(int1,
int2, ...), 371
compare(item1, item2), 371
compare(item1, item2, collation),
371
concat(string1, string2, ...), 371
contains(string1, string2), 372
current, 366
current-date(), 367
current-dateTime(), 367
current-time(), 367
data(item, ...), 366
dateTime(date, time), 367
day-from-date(date), 368
day-from-dateTime(datetime), 368
default-collation(), 367
distinct-values((item1, item2,
...), collation), 370
document, 366
document-uri(node), 366
element-available, 366
empty(item1, item2, ...), 370
end-with(string1, string2), 372
error(), 369
error(error), 369
error(error, description), 369
error(error, description, error-
object), 369
escape-uri(stringurl, escres), 372
exists(item1, item2, ...), 370
false(), 367
floor(num), 370
format-number, 366
function-available, 366
generate-id, 366
hours-from-dateTime(datetime), 368
hours-from-duration(datetime
duration), 368
hours-from-time(time), 368
implicit-timezone(), 367
index-of((item1, item2, ...),
searchitem), 370
insert-before((item1, item2, ...),
position, inserts), 370
key, 366
lang(lang), 369
last(), 367
local-name(), 369
local-name(nodeset), 369
lower-case(string), 372
matches(string, pattern), 372
minutes-from-dateTime(datetime),
368
minutes-from-duration(datetime
duration), 368
minutes-from-time(time), 368
month-from-date(date), 368
month-from-dateTime(datetime), 368
months-from-duration(datetime
duration), 368
name(), 369
name(nodeset), 369
namespace-uri(), 369

years-from-duration(datetimeduration) function (XSLT)

- namespace-uri(nodeset), 369
 - nilled(node), 366
 - node-name(node), 366
 - node-set, 366
 - normalize-space(), 371
 - normalize-space(string), 371
 - normalize-unicode(), 372
 - not(blarg), 367
 - number(arg), 370
 - position(), 367
 - remove((item1, item2, ...),
position), 370
 - replace(string, pattern,
replacement), 372
 - resolve-uri(relative, base), 372
 - reverse((item1, item2, ...)), 370
 - root(), 369
 - root(node), 369
 - round-half-to-even(num), 370
 - round(num), 370
 - seconds-from-dateTime(datetime),
368
 - seconds-from-duration(datetime
duration), 368
 - seconds-from-time(time), 368
 - starts-with(string1, string2), 372
 - static-base-uri(), 367
 - string(argument), 371
 - string-join((string1, string2,
...), separator), 371
 - string-length(), 371
 - string-length(string), 371
 - string-to-codepoints(string), 371
 - subsequence((item1, item2, ...),
start, length), 371
 - substring-after(string1, string2),
372
 - substring-before(string1,
string2), 372
 - substring(string, start), 371
 - substring(string, start, length),
371
 - system-property, 366
 - timezone-from-date(date), 368
 - timezone-from-time(time), 368
 - tokenize(string, patternstring),
372
 - trace(value, label), 369
 - translate(string1, string2,
string3), 372
 - true(), 367
 - unordered((item1, item2, ...)), 371
 - unparsed-entry-uri, 366
 - upper-case(string), 372
 - year-from-date(date), 368
 - year-from-dateTime(datetime), 368
 - years-from-duration(datetime
duration), 367
- XSS (Cross Site Scripting), 18**
- XUL (XML User Interface Language), 5**
- Y**
- year-from-date(date) **function (XSLT),
368**
 - year-from-dateTime(datetime)
function (XSLT), 368
 - years-from-duration(datetime
duration) **function (XSLT), 367**

