

Index

SYMBOL

! (exclamation point), meaning of, **23, 28**

A

Abort **method of WorkflowInstance class**
description of, 99
using with persistence services, 172

ACID transactions, 41

Actions, defining in SharePoint Designer 2007, 378

active property of rule sets, explanation of, 224

activities. See also communication activities; state-machine activities

adding to workflows, 160
CallExternalMethod, 87, 126
cleaning up names of, 29
Code, 23
Code activity, 126
CompensatableSequence, 127–128
CompensatableTransactionScope, 128–129
Compensate, 127–128
CompensationHandler, 127–128
ConditionedActivityGroup, 130
creating, 152–153
Delay, 130
determining statuses of, 86
developing iterative activities, 207–210
enabling multiple execution of, 135–136
EventDriven, 84–85, 131
EventHandlingScope and EventHandlers, 131
FaultHandler and FaultHandlers, 132–133
FaultHandlers, 83
HandleExternalEvent, 87, 133
IfElse, 27
IfElse and IfElseBranch, 133–134
InvokeWebService, 134
InvokeWorkflow, 135
Listen, 135

Parallel, 135–136

Policy, 137

renaming, 23

Replicator, 137

Sequence, 138

SetState, 138

State, 138

StateFinalization and StateInitialization, 139

Suspend, 139–140

SynchronizationScope, 140

Terminate, 141

testing, 159

Throw, 141

TransactionScope, 128–129

types of, 125

using CompositeActivity class, 146

WebServiceFault, 141

WebServiceInput, 141–142

WebServiceOutput, 142

While, 142

workflows as, 36–38

Activity **class, 36–38**

using, 145

activity classes, decorating with DesignSerializer attribute, 68

activity components

ActivityCodeGenerator class, 151–152

ActivityDesigner class, 150

ActivityValidator classes, 151

ToolboxItem class, 150

WorkflowMarkupSerializer class, 151

activity designer, views in, 85

activity execution model, advantage of, 206

activity properties

applying property attributes to, 148

binding, 147–148

DependencyObject and DependencyProperty,

146–147

types of, 147

activity rules. *See also* rules

- for ConditionedActivityGroup, 226–227
- for IfElse, 226–227
- for Policy activity, 228
- for Replicator, 226–227
- for While, 226–227
- ActivityBind **class**, **binding activities with**, 147–148
- ActivityCodeGenerator **class**, **using**, 151–152
- ActivityDesigner **class**, **using**, 150
- ActivityExecutionContext **class**, **using**, 206–207
- ActivityMarkupSerializer **class**, **description of**, 67
- ActivityTrackingLocation **class constructor**, **using**, 175
- ActivityValidator **classes**, **using**, 151
- Add Web Reference dialog box, **displaying**, 134, 336
- AddService **method**
 - exposing with workflow runtime, 95
 - using with ExternalDataExchangeService instance, 104
- administration feature in SharePoint, **description of**, 368
- alerts, **configuring in Performance Monitor**, 316
- annotations, **using with tracking profiles**, 181
- anonymous methods, **using with HelloWorld project**, 25
- App.config settings, **relationship to WCF code in OnStart method**, 353–354
- AppDomains, **relationship to WorkflowRuntime class**, 92–93
- ApplicantName-property, **using in communication service**, 114–115
- application flow interaction scenario, 6
- ApplyWorkflowChanges **method of WorkflowInstance class**, **description of**, 99
- Approval workflow in SharePoint, **description of**, 361–362
- architects, **use of Windows Workflow Foundation by**, 17
- arithmetic operators, **using in rules language**, 225
- art request process, **developing in Visual Studio**, 383–386
- ASP.NET
 - considering use of, 320–321
 - features of, 60
 - managing workflow runtime in, 321–322
- ASP.NET page flow engine
 - building, 324–331
 - communication plumbing for, 324
 - communication service for, 326
 - relationship to MVC (Model-View-Controller) pattern, 323
 - WorkflowHelper class for, 326–328
- ASP.NET runtime, **adding HTTP module to**, 342
- ASP.NET Web Service option, **selecting**, 335
- ASP.NET web services, **declaring**, 345–346. *See also* web services
- assembly resources, **adding images to**, 156–157
- association forms, **using**, 372
- associations
 - making in SharePoint 2007, 368–370
 - Start Options for, 369

- asynchronous mode, **running Windows Workflow Foundation in**, 43
- atomic property, **role in ACID transactions**, 41
- attribute-based chaining, **using with rules**, 230–231
- attributes, **applying property attributes to activity properties**, 148–149
- Avalon. *See* WPF (Windows Presentation Foundation)

B

- BAL (base activity library), **activities in**, 38
- batch services. *See also* work batch services
 - developing, 101
 - inheritance of, 166
- BCL (Base Class Library), **relevance to .NET Framework**, 12
- binding values, **using with CallExternalMethod activity**, 108
- BizTalk Server
 - mapping in, 55
 - orchestration in, 56
 - ports in, 56
 - schema in, 54–55
 - versus Windows Workflow Foundation, 56–57
- Boolean expression, **adding to HaveFirstName method**, 28–29
- BPM (business process management), **phases of**, 3
- branches, **using with IfElse activity**, 27
- breakpoints
 - adding in Visual Studio, 305
 - placing to test WaitForEmailActivity, 218
- Browse and Select a .NET Type dialog box, **accessing**, 107
- business rules. *See* rules
- button rule, **modifying for form**, 389

C

- C#
 - exception-handling block in, 83
 - modifying tracking profiles with, 180–181
 - tracking profile in, 176–177
 - web resource for, 25
- CallExternalMethod activity
 - selecting .NET interfaces for, 87–88
 - using, 126
 - using with ASP.NET page flow engine, 324
- CallExternalMethod **communication activity**, **using**, 107–110
- Cancellation & Error report
 - tabs in, 368
 - troubleshooting workflows with, 392
- cancellation handlers, **using**, 82, 302
- CardSpace, **role in .NET Framework 3.0**, 14
- catch blocks, **relationship to fault handlers**, 300
- CDATA block, **using with XAML workflow**, 63
- chaining options for rules
 - explicit chaining, 234
 - full chaining, 229–234
 - sequential (nonchained) chaining, 234

- CheckQueue **method, using with e-mail activity, 217**
/checktypes [+|-] **option in wcf.exe, description of, 49**
- child activity**
allowing execution of, 138
creating instances of, 137
- Choose Toolbox Items dialog box, tabs in, 79**
- class metadata workflow code sample, 63**
- classes**
creating for activities, 152–153
splitting in code-beside model, 64–65
- CLR (Common Language Runtime), relationship to WorkflowRuntime class, 92**
- Code activity. See also activities**
adding to web service called inside workflow, 338–339
placing in ifElseBranchActivity2 branch, 29
renaming in HelloWorld, 23, 29
using, 126–127
using in code-only workflow, 65–66
- code conditions**
advantage of rules over, 227
relationship to rules, 222
using with activities, 226–227
- code listings. See also syntax**
activity (HelloWorldActivity), 144
Activity class, 36–38
activity designer custom theme class, 150–151
ActivityExecutionContext class metadata, 206
ASP.NET web service, 345–346
batch service class customization, 190
binding for activity property, 148
BizTalk schema map, 55
Boolean expression added to HaveFirstName method, 28–29
CallExternalMethod activity syntax, 126
class metadata workflow, 63
code conditions, 222
CodeActivity instance for dynamic update, 277
code-behind for communication service, 122
code-beside class for ASP.NET page flow engine, 330
code-only workflow, 65–66
college application communication service workflow, 114–115
communication contract, 350–351
communication contract implemented as .NET interface, 112
communication interface for correlation, 121
communication plumbing for page flow engine, 324
communication service, 113
communication service for page flow engine, 326
communication service implementation, 121–122
CompensatableSequence activity syntax, 127
CompensatableTransactionScope activity, 128–129
Compensate activity syntax, 127
CompensationHandler activity syntax, 127
CompiledAssembly property, 71
CreateWorkflow method used with rules, 247–248
CreateWorkflow method with Type instance, 94
data event arguments class for college application communication service, 112–113
database created for tracking service, 182
debugging workflows, 307–308
declarative rule conditions, 223
DefaultWorkflowSchedulerService, 165
dependency properties, 147
DesignSerializer attribute, 68
Dictionary<string, object> collection for HelloWorld, 26
dynamic update from within workflow, 276
e-mail activity, 214–215, 216–217
EmailTrackingChannel class, 200–201
endpoint in App.config or Web.config file, 348
EnrollmentApproved, 116–117
EnrollmentRejected event and wca.exe, 118–119
EnsureSaved method for workflow designer hosting, 286
event handler for workflow designer hosting, 286–287
exception handler, 296
exception throwing, 295
exception-handling block in C#, 83
Execute method of WaitForEmailActivity, 217–218
explicit chaining, 233–234
ExternalDataEventArgs class, 105–106
ExternalDataExchangeAttribute, 105
ExternalDataExchangeService instance with AddService method, 104
feature.xml file for SharePoint, 380–381
file persistence service, 191–196
FileWorkflowPersistenceService, 191–197
ForEachActivity class, 207–209
full chaining options for rules, 229
GetAllServices method, 95
GetService method used with runtime service, 95
GetWorkflowDefinition method of WorkflowInstance class, 274
Global.asax with Application_Start and Application_End event handlers, 321
HelloWorld with comment, 30
HelloWorldActivity, 144
hosting workflow designer Windows Forms fields and constructor, 283–284
IEventActivity interface, 211
IfElse activity method, 28
InitializeComponent method generated by workflow compiler, 64
Initialized event for Replicator activity, 385
install.bat file added to SharePoint for Visual Basic, 382–383
InvokeWorkflowActivity instance for dynamic update, 274–275
IWebService interface, 341
layout data loaded from filesystem, 292
ListView control for workflow designer hosting, 284–285
local communication via .NET interface, 40

code listings (continued)

ManualWorkflowSchedulerService, 166, 320
message in console for HelloWorld, 26
message retrieved from dictionary, 26
MethodInvoking event handler, 385
MyActivityToolboxItem class, 154–155
MyWorkflow class and `x:Class` attribute, 65
.NET code behind workflow, 21–22
parameters for workflow communication, 102–103
partial keyword, 65
PatientRecord class for web service, 336
PendingWorkItem class, 196–197
PersistOnClose attribute, 172
Program.cs file for HelloWorld project, 24–25
property promotion for DependencyProperty, 109
RemoveService method for workflow runtime, 95
RequestEnrollmentStatus class, 115–116
rule sets, 224
RuleConditionDialog UI, 237
RuleDefinitions object serialization, 242–244
RuleInvoke chaining, 233
RuleRead chaining, 231
rules updated dynamically, 248–249
rules XML file for FreeShipping rule, 45–46
RuleSetDialog class, 238
RuleWrite chaining, 231–232
runtime classes enabled in host application, 94
runtime event handling, 97–98
save and close methods for workflow designer hosting, 287
scheduler for web service, 343
serialization, 66–67
shopping cart workflow application, 264–265
shopping cart workflow Windows Forms application, 266–270
SQL tracking query, 185
SQL tracking service added to workflow runtime, 51
SqlWorkflowPersistenceService, 171
state-machine workflow, 290–291
StateMachineWorkflowInstance class, 260
TerminationTrackingService, 188
ThankYou.aspx for ASP.NET page flow engine, 330
throwing faults, 303
ToolboxItemAttribute, 79
ToolboxItemAttribute for college application communication service, 115–116
trace sources, 312
trace switch, 310
tracing, 308
tracking profile in C#, 176–177
tracking profile in XML, 42, 177–178
tracking profile serialization, 178
tracking query, 185
tracking service instance of service class, 183
TrackingService, 198–200
TransactionScope activity, 128–129
UpdateTrackingProfile procedure, 184
WaitForEmailService, 213–214
wca.exe data-exchange service, 117–119

WCF service contract, 347
web service called inside workflow, 335
Web.config file for WorkflowRuntime class, 322
WebServiceInputActivity class, 152
wfc.exe compiler, 68–69
work batch service, 166–167
workflow communication contract, 350
workflow compilation with wfc.exe, 68
workflow definition class for workflow communication, 103
workflow host code for communication service, 123
workflow host for college application communication service, 111–112
workflow host for faults, 297–298
workflow loader service, 202
workflow queue, 212
workflow runtime for web service, 344
workflow runtime service class customization, 203
workflow with code, 9
Workflow1.designer.cs file contents, 73–74
WorkflowCompiler class, 70
WorkflowCompleted event handler, 345
WorkflowDesignerLoader class in designer API, 279–280
WorkflowHelper class, 329
WorkflowHelper class for page flow engine, 326–328
WorkflowHostService, 352–353
WorkflowQueue class, 211
WorkflowRuntime class reference, 92
WorkflowSchedulerService, 190
workflow.xml file for SharePoint, 381
WriteFileActivity class, 153–154
WriteFileActivityDesigner class implementation, 155–156
WriteFileActivityDesigner class skeleton, 155
WriteFileActivityDesignerTheme class, 157
WriteFileActivityValidator class, 158–159
XAML (Extensible Application Markup Language), 13
XAML workflow, 62
`x:Class` attribute and MyWorkflow class, 65
XML rental rules, 245–246
XML structure for form, 388
XmlWriter instance for database, 292
CodeActivity **instance for dynamic update code sample, 277**
code-beside class
for exposing workflows as web services, 340
using with ASP.NET page flow engine, 330
for web service called inside workflow, 337–338
code-beside model
using, 64–66
using with HelloWorld project, 21
using with iterative activities, 210
CodeDom
comparison classes of, 240–241
expression classes of, 239
relationship to rules, 238–241
code-only model, using, 65–66

- Collect Feedback workflow in SharePoint, description of, 362–363**
 - Collect Signatures workflow in SharePoint, description of, 363**
 - college application college application communication service, 110–115**
 - expanding, 115–117
 - college application communication service, 110–115**
 - command-line utilities**
 - wca.exe data-exchange service, 48
 - wfc.exe compiler, 48
 - comments**
 - adding to HelloWorld project, 30
 - obtaining with Collect Feedback workflow, 362–363
 - commit batch services, maintaining integrity with, 41**
 - Commit method, using with work batch services, 166–167**
 - Common Language Runtime (CLR), relationship to WorkflowRuntime class, 92**
 - communication. See workflow communication**
 - communication activities. See also activities**
 - CallExternalMethod, 107–110
 - customizing, 115–117
 - generating, 117–119
 - HandleExternalEvent, 110
 - Listen and EventDriven, 110
 - communication contract code sample, 350–351**
 - communication contracts**
 - defining, 104
 - implementing as .NET interface, 112
 - communication plumbing for page flow code, 324**
 - communication service interface**
 - implementing for page flow engine, 326
 - using CorrelationAliasAttribute with, 121
 - using CorrelationInitializerAttribute with, 120
 - using CorrelationParameterAttribute with, 120
 - comparison classes of CodeDom, descriptions of, 240–241**
 - CompensatableSequence activity, using, 127–128**
 - CompensatableTransactionScope activity, using, 128–129**
 - Compensate activity, using, 127–128**
 - compensation logic, initiating, 128**
 - CompensationHandler activity, using, 127–128**
 - compilation**
 - performing with wfc.exe, 68–70
 - performing with WorkflowCompiler class, 70–71
 - steps for, 71
 - CompiledAssembly property, exposing, 71**
 - CompletedScope table, using with SqlWorkflowPersistenceService, 170**
 - composite activities, explanation of, 125**
 - CompositeActivity class, using, 146**
 - CompositeActivityMarkupSerializer class, description of, 67**
 - condition, providing for IfElse activity, 28**
 - Condition property**
 - expanding for HelloWorld, 29
 - using with IfElseBranch activity, 133–134
 - conditional code, using with DynamicUpdate-Condition property, 277**
 - conditional expressions, specifying, 226–227**
 - conditional logic flow, activities for, 133–134**
 - ConditionedActivityGroup activity**
 - using, 130
 - using activity rules with, 226–227
 - conditions, defining in SharePoint Designer 2007, 378**
 - configuration errors, checking at design time, 86**
 - configuration files, examples of, 50–51**
 - connected systems. See WCF (Windows Communication Foundation)**
 - consistent property, role in ACID transactions, 41**
 - console application, creating for HelloWorld project, 24–27**
 - content, managing expired content, 364–365**
 - content types, associating workflows to, 369**
 - context menu, using with state machine workflow designer, 257–258**
 - controller in MVC pattern, description of, 323**
 - correlation**
 - example of, 121–124
 - process of, 119
 - correlation parameter, using, 124**
 - CorrelationAliasAttribute, using with communication service interface, 121**
 - CorrelationInitializerAttribute, using with communication service interface, 120**
 - CorrelationParameterAttribute, using with communication service definition interface, 120–121**
 - counter data, capturing with Performance Monitor, 315**
 - createMessage property, using in HelloWorld project, 23**
 - createMessage2 activity, using with HelloWorld, 29**
 - CreateWorkflow method**
 - calling, 99
 - calling to start workflow instances, 94
 - code sample, 247–248
 - passing parameters to, 102
 - CurrentState property, using, 261**
 - CurrentStateName property, using, 261**
 - custom workflows**
 - debugging in SharePoint, 391–392
 - deploying and configuring to SharePoint, 390–391
- ## D
- data entry forms, modeling with InfoPath, 58**
 - data exchange service, features of, 189**
 - data source, modifying for form, 388**
 - databases**
 - creating for tracking services, 182
 - creating for workflow persistence data store, 169
 - writing XmlWriter instance for, 292
 - data-exchange services, using, 48**

/debug[+|-] option in wcf.exe, description of

/debug[+|-] option in wcf.exe, description of, 48 debugging

custom workflows in SharePoint, 391–392
in Visual Studio, 87

debugging workflows. *See also* tracing

with logs and alerts, 315–316
with performance counters, 312–315
with tracing, 308–312
in Visual Studio, 304–306
with `WorkflowRuntime` events, 306–308

declarative rules, 223

versus rule sets, 223
using with activities, 226–227

`DefaultWorkflowCommitWorkBatchService`,
using, 167

`DefaultWorkflowSchedulerService`, using, 165

Delay activity, using, 130

/delaysign[+|-] option in wcf.exe, description of,
48

dependency properties, using with `CallExternal- Method` activity, 109

`DependencyObject` and `DependencyProperty`
classes, using with activities, 146–147

`DependencyProperty`, using with college application
communication service, 115–116

design hosting, features of, 49–50

design phase of BPM, explanation of, 3

designer API

`DesignSurface` class in designer API, 280–281
`WorkflowDesignerLoader` class in, 278–280
`WorkflowView` class in designer API, 281

designer re-hosting

capabilities of, 278
using, 293

designer services, using, 281–282

designer-related services, descriptions of, 280–281

designers. *See* workflow designers

`DesignSerializer` attribute, decorating activity
class with, 68

`DesignSurface` class in designer API, using, 280–281

design-time validation, role in Visual Studio, 86

dictionary, retrieving messages from, 25–26

`Dictionary<string, object>` collection

using in workflow communication, 102
using with HelloWorld project, 26

digitally signing documents, performing with `Collect Signatures` workflow, 364

dirty workflow, indicating, 285

`Disposition Approval` workflow in SharePoint,
description of, 364–365

document approval, managing with `Approval` workflow,
361–362

drag-and-drop functionality, adding to workflow designer
hosting, 288–289

durable property, role in ACID transactions, 41

dynamic updates, 49

deciding on, 276–278
using, 274–275, 293
from within workflow, 275–276

`DynamicUpdateCondition` property

conditional code for, 277
using with dynamic updates, 276–277

E

EAI (Enterprise Application Integration), relationship to
`BizTalk Server`, 53

e-mail, monitoring inbox for, 213–219

e-mail activity, testing execution of, 218–219

`EmailTrackingChannel` class code sample,
200–201

Empty Workflow Project project template, description
of, 75

endpoints in WCF

components of, 347
relationship to service node, 349

`EnqueueItem` method of `WorkflowInstance` class,
description of, 100

`EnqueueItemOnIdle` method of `WorkflowInstance`
class, description of, 100

`EnrollmentApproved` code sample, 116–117

`EnrollmentService` instance, including in college
application communication service, 111–113

`EnsureSaved` method for workflow designer hosting,
286

error messages

displaying for HelloWorld project, 23
identifying, 28

event activities

developing, 213–219
`IEventActivity` interface of, 211
workflow queues, 211–212

event handlers

for ASP.NET page flow engine, 328
for `CodeActivity` of dynamic update, 277
configuring from Properties window, 81
delivering messages to, 119
using with shopping cart workflow application, 271
for workflow designer hosting, 285–287, 289–290

EventDriven activities

using, 84–85, 110, 131, 254
using with state machine workflow designer, 258, 260

event-driven workflows, support for, 5–6, 8, 16–17

`EventDrivenActivity` state-machine activity class,
description of, 39

`EventHandlingScope` and `EventHandlers` activities,
using, 131

events

accessing, 81
listening for, 135

`Events` property of `WorkflowTrackingLocation`,
valid values for, 176

events view, displaying for Properties window, 81

Excel, new features in, 357

Excel Services, features of, 358

exception handlers, specifying, 296

exceptions
 derivation of, 295
 versus faults, 296
 throwing, 295

exclamation point (!), meaning of, 23, 28

Execute method
 using with `WaitForEmailActivity`, 217–218
 using with `WriteFileActivity` class, 154

ExecuteCode event handler
 defining in code-only model, 65–66
 modifying for web service called inside workflow, 338
 using in college application communication service, 114–115

ExecuteCode event, using with Code activity, 126–127

ExecuteCode property, setting for HelloWorld project, 24

execution path, modifying with dynamic update, 49

execution/deployment phase of BPM, explanation of, 3

ExecutionStatusEvents property, using with ActivityTrackingLocation class, 175

expired content, managing with Disposition Approval workflow, 364–365

explicit chaining, using with rules, 233–234

expression classes for CodeDom, descriptions of, 239

expression joiners, using in rules language, 226

expressions, using in rules language, 225

extensibility, significance of, 50

Extensible Application Markup Language (XAML)
 persisting workflow definitions to, 66–68
 relationship to WPF, 12–13

extensible models, basis of workflows on, 4–5

external web services, making calls to, 134

ExternalDataAttribute, using with communication-service interfaces, 105

ExternalDataEventArgs class
 properties in, 106–107
 using with local communication services, 105–107

ExternalDataExchange runtime service
 public methods exposed by, 105
 using with communication service classes, 104

ExternalDataExchangeService instance, in college application communication service, 111–112

F

F5, pressing to test HelloWorld project, 27

fault handlers
 accessing, 299
 versus cancellation handlers, 302
 placing in workflows, 301

Fault property, using with Throw activity, 148

FaultHandler activities
 adding to `FaultHandlers` activity, 83, 88
 using, 132–133

FaultHandlers activities, using, 132–133, 299

faults
 analyzing messages related to, 298
 versus exceptions, 296

explanation of, 296–299
 handling, 299–302
 raising declaratively, 303
 throwing, 302–304

faults view, description of, 83

feature.xml file for SharePoint, 380–381

field data types, choosing in SharePoint Designer, 376

Figures
 Activity class, 145
 activity designer, 85
 Add New Item dialog box, 80
 Add submenu for Project menu, 80
 Add Web Reference dialog box, 134, 336
 Approval workflow in SharePoint, 361
 BAL (base activity library) activities, 38
 binding dialog box for web service called inside workflow, 339
 BizTalk Server mapping interface, 55
 BizTalk Server related to hub-and-spoke model, 54
 breakpoint during execution, 306
 Breakpoint option in Visual Studio, 305
 Browse and Select a .NET Type dialog box, 107
 CancellationHandler default view, 83
 Collect Feedback workflow in SharePoint, 363
 Collect Signatures workflow in SharePoint, 364
 communication workflow in Visual Studio, 111
 ConditionedActivityGroup with Code activity, 146
 context menu in state machine workflow designer, 258
 debugging functionality in Visual Studio, 304
 Disposition Approval workflow in SharePoint, 365
 e-mail activity, 219
 event-driven workflow, 17
 faults view for fault handler catching `SqlException`, 84
 Faults view of workflow, 132
`forEachDate` method of `ForEachActivity` class, 209
 HelloWorld project, 27
 HelloWorld workflow, 30
 identity card issued in Windows Vista, 14
`IfExists` activity, 27
`IfExists` activity before configuration, 28
 InfoPath form, 59
 instructions for IT recruiter for hiring process, 7
`InvalidOrder` state for shopping cart workflow application, 264
 Microsoft Excel, 357
 Microsoft Word 2007, 356
 .NET execution model, 93
 .NET Framework stack, 11
 New Project dialog box, 20
 New Project dialog box in Visual Studio, 72
 ordered workflow, 16
 page flow engine, 325
 Parallel activity functions, 136
 Performance Monitor, 51, 313
`Program.cs` and `Workflow1.cs` files, 73
 Project menu, 77
 Properties window in Visual Studio, 81–82

Figures (continued)

- property binding dialog box in Visual Studio, 148
- recursive state composition, 255–256
- report in SharePoint, 368
- Rule Condition Editor, 237
- Rule Set Editor dialog box, 236
- rules file associated with workflow file, 245
- RuleSet Editor, 45
- sequential workflow, 21, 35
- setTheState SetState activity, 139
- SharePoint, 58
- SharePoint workflow in Visual Studio, 385
- Shopping Cart Workflow application, 263
- state machine for shopping cart workflow, 263
- state machine workflow designer, 257, 259
- state-machine workflow, 290
- Theme Customization dialog box, 87
- Three-state workflow's association screen, 372
- toolbar, 77
- tracking infrastructure, 180
- tracking software bugs, 8
- Visual Studio 2005 development screen, 47
- Visual Studio Toolbox, 78
- Visual Studio Toolbox displaying custom components, 78
- WaitForEmailActivity, 218
- While activity, 143
- workflow designer hosting, 288
- Workflow Designer wizard in SharePoint, 376
- workflow development view, 84
- workflow implementation for exposing workflow as windows service, 350
- Workflow menu, 76
- workflow stack, 34
- Workflow Status screen in SharePoint, 367
- XAML form, 13
- XAML workflow in workflow designer, 63
- file persistence service code sample, 191–196**
- filesystem**
 - loading layout data from, 292
 - working with workflow definitions in, 292
- FileWorkflowPersistenceService **code sample, 191–197**
- first names, passing to workflows, 26**
- ForEachActivity **class, using with iterative activities, 207–209**
- forms. See also Windows Form shopping cart workflow application; workflow forms**
 - adding submit action to, 389
 - designing, 389
 - implementing using InfoPath and Forms Server, 387–390
 - modifying button rule for, 389
 - modifying data source for, 388
- Forms Server, features of, 358**
- FreeShipping **rule, rules XML file for, 45–46**
- FrontPage. See SharePoint Designer 2007**
- FSMs (finite state machines), support for, 5–6, 8, 16–17**

full chaining options for rules

- attribute-based chaining, 230–231
- implicit chaining, 230
- process of, 229–230
- RuleInvoke chaining, 232–233
- RuleRead chaining, 231
- RuleWrite chaining, 231–232

G

- generic, OutputParameters property as, 25**
- get **accessor**
 - using in workflow communication, 103
 - using with WriteFileActivity class, 153
- GetAllServices **method, using with runtime services, 95**
- GetService **method**
 - using with ExternalDataExchange service, 105
 - using with runtime services, 95
- GetWorkflowDefinition **method**
 - of WorkflowInstance class, 274
 - of WorkflowInstance class, description of, 100
- GetWorkflowQueueData **method of WorkflowInstance class, description of, 100**
- Global class**
 - accessing global variables in, 346
 - using with web services, 344
- Global.asax with Application_Start and Application_End event handlers, 321**
- glyphs, relationship to designer API, 282**

H

- HandleExternalEvent activity**
 - selecting .NET interfaces for, 87–88
 - using, 110, 133
 - using with ASP.NET page flow engine, 324
- HaveFirstName **method, using with HelloWorld, 28**
- HelloWorld project**
 - adding decision-making logic to, 27–30
 - commenting code in, 30
 - console application for, 24–27
 - creating project for, 20
 - naming, 20–21
 - objectives of, 20
 - requirements for, 19–20
 - workflow for, 21–24
- /help **option in wcf.exe, description of, 49**
- history feature in SharePoint, description of, 367**
- History List in SharePoint, description of, 369**
- host application**
 - enabling runtime classes in, 94–95
 - starting workflows in, 40
- hosting workflow designer. See workflow designer hosting**
- hosts. See also re-hosting; workflow designer hosting**
 - communication with, 39–40, 48
 - examples of, 39
 - SharePoint as, 360
- host-to-workflow communication, role of workflow queues in, 211–212**

HTTP module, adding to ASP.NET runtime, 342
hub-and-spoke model, describing BizTalk Server in terms of, 54
human-to-system interaction scenario, 6

I

IActivityEventListener, using with e-mail activity, 216
identity cards, use in Windows CardSpace, 14
Identity property in ExternalDataEventArgs class, description of, 106
IDesignerGlyphProviderService interface, using, 282
IEventActivity interface
 using, 213
 using with Delay activity, 130
 using with event activities, 211
IfElse activity
 locating, 27
 providing condition for, 28
 using, 133–134
 using activity rules with, 226–227
 using branches with, 27
IfElseBranch activity
 accessing fault handlers for, 299
 using, 133–134
if-then-else statements, conceptualizing rules as, 44
iisreset, performing for custom workflows in SharePoint, 390–391
IL (Intermediate Language), relevance to .NET Framework, 11–12
images, adding to assembly resources, 156–157
IMemberCreationService, using, 282
implicit chaining, using with rules, 230
inbox, monitoring for e-mail, 213–219
Indigo. See WCF (Windows Communication Foundation)
InfoPath
 features of, 58, 358
 implementing forms with, 387–390
InitializeComponent method generated by workflow compiler, code listings, 64
initiation forms, using, 373
install.bat file, adding to SharePoint for Visual Basic, 382
instance activity properties, using, 147
instance locking, support in persistence services, 196 instances. See workflow instances
InstanceState table, columns in, 169–170
InterfaceType property, using with CallExternalMethod activity, 107
InvokeWebService activity
 configuring, 337
 description of, 334
 using, 134
 using with web service called inside workflow, 338
InvokeWorkflow activity, using, 135

InvokeWorkflowActivity instance, adding for dynamic update, 274–275
isolated property, role in ACID transactions, 41
ISVs (Independent Software Vendors), use of Windows Workflow Foundation by, 18
Iterate method, using with ForEachActivity class, 209
iterative activities, developing, 207–210
IWebService interface, creating, 341

K

K2.net web address, 10
/keycontainer:<string> option in wcf.exe, description of, 48
/keyfile:<file> option in wcf.exe, description of, 48
KPIs (key performance indicators), relationship to tracking services, 174

L

/language:[cs|vb] option in wcf.exe, description of, 49
/library:<path list> option in wcf.exe, description of, 48
Listen communication activity, using, 110, 135
ListView control
 subscribing to MouseMove event of, 288–289
 using with workflow designer hosting, 284–285
Load method of WorkflowInstance class, description of, 100
LoadStateMachineLayout method, using with state-machine designer, 293
LOB (line of business) systems, features of, 9
local communication services. See also workflow communication
 classes related to, 104–107
 examples of, 110–115
 relationship to hosting, 40
logs, configuring in Performance Monitor, 315
long-running transactions, 42
 compensating for, 129
 providing functionality for, 127
Lutz's Reflector, using with XAML workflow, 63

M

Main method, event handlers in, 97–98
ManualWorkflowSchedulerService
 code sample, 320
 using, 165–166
mapping, role in BizTalk Server, 55
markup only workflow style, using, 61–64
markup-only model, using with wcf.exe, 68
menus
 Project menu, 76
 Workflow menu, 75–76
Message property, using in HelloWorld project, 22

messages

- analyzing for faults, 298
 - categorizing with `TraceSwitch` class, 310
 - delivering to event handlers, 119
 - displaying in console for `HelloWorld`, 26–27
 - implementing tracing messages, 312
 - retrieving from dictionary, 25
 - retrieving from workflows, 25
 - role in `BizTalk Server`, 54–55
 - turning on and off in tracing infrastructure, 309
- meta activity properties, using, 147**
- `MethodInvoking` event handler, 385
- `MethodName` property, using with `CallExternalMethod` activity, 108
- Microsoft Office, Word and Excel features in, 356–357**
- Microsoft SharePoint 2007. See SharePoint 2007**
- model in MVC pattern, description of, 323**
- modification forms, using, 374**
- monitoring, role in BPM, 3**
- `MouseMove` event of `ListView`, subscribing to, 288–289
- MVC (Model-View-Controller) pattern, relationship to ASP.NET page flow engine, 323**
- `MyActivityToolboxItem` class code sample, 154–155
- `MyParameter` property, using with `CallExternalMethod` communication activity, 110
- `MyWorkflow` class, creating in code-beside model, 65

N

- namespaces, availability of, 52–53
- .NET code, displaying for workflows, 21–22**
- .NET developers, use of Windows Workflow Foundation by, 16–17**
- .NET exception handlers, throwing from .NET code, 302–303**
- .NET execution model, 93**
- .NET Framework 3.0**
 - using with `HelloWorld` project, 19–20
 - WCF (Windows Communication Foundation) component of, 13
 - Windows CardSpace component of, 14
 - WPF (Windows Presentation Foundation) component of, 12–13
- .NET Framework, features of, 11–12**
- .NET interface, using for local communication, 40**
- .NET transactions, resource for, 129**
- New Project dialog box, displaying, 20**
- `/nocode[+|-]` option in `wcf.exe`, description of, 49
- `/nologo` option in `wcf.exe`, description of, 49
- `/nowarn` option in `wcf.exe`, description of, 49

O

- objects, referencing with rules, 225
- Office suite, new features in, 358**
- `OnEvent` method, using with e-mail activity, 217
- `OnStart` method, WCF code in, 353–354
- optimization phase of BPM, explanation of, 3**
- orchestration, role in BizTalk Server, 56**

- ordered workflows, support for, 5, 16**
- `/out:<file>` option in `wcf.exe`, description of, 48
- output parameters, accessing for workflow communication, 102–103**
- `OutputParameters`, using with `HelloWorld` project, 25

P

- page flow engine. See ASP.NET page flow engine**
- Parallel activity, using, 135–136**
- parallel execution, impact of SynchronizationScope activity on, 140**
- parameters**
 - using correlation parameter in workflow communication, 124
 - using for workflow communication, 102–103
- partial classes**
 - declaring .NET code files as, 65
 - using with code-beside model, 64–65
 - in `Workflow1.designer.cs` file, 73
- partial keyword code sample, 65**
- Passport versus Windows CardSpace, 14**
- `PatientRecord` class for web service, 336
- `PendingWorkItem` class code sample, 196–197
- performance counters**
 - availability of, 51–52
 - using, 312–315
- Performance Monitor**
 - capturing counter data with, 315
 - configuring alerts in, 316
- persistence points, occurrence of, 99, 172**
- persistence services, 42**
 - code sample, 191–196
 - developing, 191–197
 - features of, 168
 - `SqlWorkflowPersistenceService`, 168–171
- `PersistOnClose` attribute code sample, 172
- Policy activity**
 - using, 137
 - using rule sets with, 228
- portals. See SharePoint 2007**
- ports, role in BizTalk Server, 56**
- `PossibleStateTransitions` property, using, 261
- priority property of rule sets, explanation of, 223**
- profiles. See tracking profiles**
- Program.cs file**
 - contents of, 24–25, 73
 - opening for `HelloWorld` project, 24
- project item types, adding in Visual Studio, 80–81**
- Project menu, options on, 76**
- project templates**
 - Empty Workflow Project, 75
 - Sequential Workflow Console Application, 73–74
 - Sequential Workflow Library, 75
 - State Machine Workflow Library, 75
 - State-Machine Workflow Console Application, 75
 - Workflow Activity Library, 75
- projects**
 - creating for `HelloWorld` example, 20
 - testing, 27

projects in Visual Studio

- adding to solutions, 73
- creating, 72–73
- output types for, 72
- types of, 73–75

Properties window in Visual Studio, components of, 81–82**property attributes, applying to activity properties, 148–149****property binding, performing for activities, 147–148**

`ProxyClass` **property, setting for InvokeWebService activity, 134**

Q

queues. *See* **workflow queues**

R**recursive state composition**

- relationship to `State` activity, 253
- using, 255–256

red exclamation point (!), meaning of, 23, 28**reevaluation property of rule sets, explanation of, 223**

`/reference:file list>` **option in wcf.exe, description of, 48**

re-hosting. *See also* **hosts; workflow designer hosting**

- capabilities of, 278
- using, 293

`ReloadTrackingProfiles` **method of WorkflowInstance class, description of, 100**

RemoveService method

- using with `ExternalDataExchange` service, 105
- using with workflow runtime, 95

Replicator activity

- Initialized event for, 385
- using, 137
- using activity rules with, 226–227

reporting feature in SharePoint, description of, 368

`RequestEnrollmentStatus`-**class code sample, 115–116**

`RequiredItemEventArgs` **instances, using with communication service, 122**

`Resume` **method of WorkflowInstance class, description of, 100**

`/rootnamespace:<string>` **option in wcf.exe, description of, 49**

Rule Condition Editor, using, 236**rule definitions, storage of, 45****rule evaluation.** *See* **chaining options for rules****rule reevaluation option, explanation of, 235****Rule Set Editor**

- accessing, 44
- using, 235–236

rule sets, 223–224

- versus declarative rules, 223
- properties of, 223
- using with `Policy` activity, 228

`RuleConditionDialog` **UI, using, 237**

`RuleConditionReference`, **specifying conditional expressions with, 226–227**

`RuleDefinitions` **object serialization code sample, 242–244**

`RuleInvoke` **chaining, using with rules, 232–233**

`RuleRead` **chaining, using with rules, 231**

rules. *See also* **activity rules**

- advantage over code conditions, 227
- class support for, 222
- conceptualizing, 44
- declarative rules, 223
- defining, 224–226
- evaluating with `Policy` activity, 137
- programmatic example of, 241–245
- referencing objects with, 225
- relationship to code conditions, 222
- relationship to `CodeDom`, 238–241
- updating dynamically, 248–249
- using, 247–248
- in XML, 245–247

rules API, features of, 241**rules editor, creating, 245****rules language**

- arithmetic operators used in, 225
- expression joiners used in, 226
- expressions used in, 225

rules UI

- embedding in custom applications, 237–238
- features of, 235

rules XML file for FreeShipping rule, 45–46

`RuleSetDialog` **class code sample, 238**

`RuleWrite` **chaining, using with rules, 231–232**

runtime classes, enabling in host applications, 94**runtime engine, relationship to hosting, 39****runtime events, handling, 96–98****runtime services.** *See also* **workflow runtime**

- ACID transactions, 41
- custom services, 44
- data exchange service, 189
- developing, 203
- developing for event activity, 213–214
- developing persistence services, 191–197
- developing scheduling services, 190
- developing tracking services, 198–202
- developing work batch services, 190
- developing workflow loader services, 202
- listing, 95
- long-running transactions, 42
- managing, 94–95
- obtaining references to, 95
- persistence services, 42
- removing, 95
- scheduling services, 43–44, 164–166
- tracking services, 42–43
- transaction services, 41
- work batch services, 166–167
- workflow loader service, 189

S

save and close methods for workflow designer hosting, 287

scheduling services

- adding to workflows, 343
- DefaultWorkflowSchedulerService, 165
- developing, 190
- features of, 164
- ManualWorkflowSchedulerService, 165–166
- using, 43–44

schema, role in BizTalk Server, 54–55

Sequence activity

- description of, 85
- using, 138

sequential (nonchained) chaining, using with rules, 234

Sequential Workflow Console Application option, selecting, 20–21

Sequential Workflow Console Application template, creating projects from, 73–74

Sequential Workflow Library project template, description of, 75

sequential workflows

- accessing fault handlers for, 299
- in code-only workflow, 65–66
- example of, 16
- identifying, 35–36
- versus state-machine workflows, 36
- using EventDriven activity with, 131

sequential workflows view designer, description of, 82

Serializable attribute, using with External-DataEventArgs class, 106

serialization, 66–68

service providers, use of Windows Workflow Foundation by, 18

ServiceBase class, location of, 353

services-oriented architecture (SOA)

- versus connected systems, 333
- explanation of, 334
- relationship to WCF, 13

session between web service client and web service, managing, 342

set accessor

- using in workflow communication, 103
- using with FirstName class in HelloWorld project, 22
- using with WriteFileActivity class, 153

SetState activity, using, 138, 254

SetState method, using, 261–262

SetStateActivity state-machine activity class, description of, 39

Settings class, using with web service called inside workflow, 342

SharedConnectionWorkflowCommitWorkBatch-Service, using, 167

SharePoint 2007

- debugging custom workflows in, 391–392
- deploying and configuring custom workflows to, 390–391
- enabling workflows in, 361

features of, 358–359

as host, 360

integration with Windows Workflow Foundation, 57–59

workflow associations in, 368–370

SharePoint Designer 2007

choosing field data types in, 376

data types for variables in, 377

defining Actions in, 378

defining conditions in, 378

developing in, 375–379

SharePoint workflow features

administration, 368

history, 367

reporting, 368

tasks, 366–367, 369

SharePoint workflow project templates, installing for Visual Studio, 380

SharePoint workflows. See also SharePoint 2007; workflows

Approval, 361–362

associating to content types, 369

Collect Feedback, 362–363

Collect Signatures, 363

deploying to servers, 390–391

developing in Visual Studio, 383–386

Disposition Approval, 364–365

enabling, 361

running, 371

Three-state, 366

Translation Management, 365–366

using WorkflowInvoked event handler with, 384

SharePoint-specific tasks, adding to Visual Studio Toolbox, 383

shopping cart workflow application

description of, 262–263

state machine for, 263–264

Site Collection Features page, accessing, 390

Skelta Workflow.NET web address, 10–11

SOA (services-oriented architecture)

versus connected systems, 333

explanation of, 334

relationship to WCF, 13

solutions in Visual Studio

adding projects to, 73

contents of, 71–72

creating, 72–73

creating empty solutions, 72

SomeMessage input parameter, using in workflow communication, 103

SomeOtherMessage output parameter, using in workflow communication, 103

SQL persistence service, preparing, 169–171

SQL Server Management Studio, downloading, 169

SQL tracking query code sample, 185

SQL tracking service

adding to workflow runtime, 50–51

using, 182–183

- `SQLException` **class variable, relationship to Fault property, 133**
- `SqlTrackingService` **class. See also tracking services**
- data maintenance of, 186
 - features of, 181–182
 - preparing, 182–183
 - profiles used with, 183–184
 - querying, 184–186
 - using with tracking services, 42
 - using Workflow Monitor with, 186–187
- `SqlWorkflowPersistenceService` **class**
- and delays, 173
 - description of, 168–169
 - preparing, 169–171
 - relationship to persistence services, 42
 - using, 171
- standard activities, explanation of, 125**
- `Start` **method of WorkflowInstance class**
- calling, 100
 - description of, 100
- Start Options, using with workflow associations, 369**
- `StartRuntime` **public method, using with WorkflowRuntime class, 93–94**
- state, setting current state of workflow instances, 261–262**
- State activity, using, 138, 253–254**
- state machine implementation**
- example of, 16–17
 - guidelines for use of, 252–253
 - using with shopping cart workflow, 263–264
- state machine workflow designer, using, 257–260**
- State Machine Workflow Library project template, description of, 75**
- `StateActivity` **state-machine activity class, description of, 39**
- StateFinalization activity, using, 139, 254**
- `StateFinalizationActivity` **state-machine activity class, description of, 39**
- StateInitialization activity, using, 139, 254**
- `StateInitializationActivity` **state-machine activity class, description of, 39**
- state-machine activities. See also activities**
- EventDriven, 254
 - SetState, 254
 - State, 253–254
 - StateFinalization, 254
 - StateInitialization, 254
 - `StateMachineWorkflowActivity` class, 253
- state-machine activity classes, 39**
- state-machine designer, using, 290–293**
- state-machine instances, querying, 261**
- State-Machine Workflow Console Application project template, description of, 75**
- state-machine workflows. See also Three-state workflow in SharePoint**
- versus sequential workflows, 36
 - using SetState activity with, 138
 - using State activity with, 138
- state-machine workflows, using EventDriven activity with, 131**
- state-machine workflows view designer, description of, 84–85**
- `StateMachineWorkflowActivity` **class, using, 253**
- `StateMachineWorkflowInstance` **class**
- flexibility of, 261–262
 - information provided by, 261
 - using, 260–261
- states versus transitions, 251–252**
- `static void Main(string[] args)` **signature, 73**
- `StopRuntime` **public method, using with WorkflowRuntime class, 93**
- stored procedures, including in SQL persistence setup, 170**
- `string` **variable, using with HelloWorld project, 25**
- `StringWriter` **instance, using with tracking profile, 178**
- submit action, adding to form, 389**
- Subscribe method, using with e-mail activity, 216**
- Suspend activity, using, 139–140**
- `Suspend` **method of WorkflowInstance class, description of, 100**
- SynchronizationScope activity, using, 140**
- syntax. See also code listings**
- of `CallExternalMethod` activity, 126
 - of `Code` activity, 126
 - of `CompensatableSequence` activity, 127
 - of `CompensatableTransactionScope` activity, 128
 - of `Compensate` activity, 127
 - of `CompensationHandler` activity, 127
 - of `ConditionedActivityGroup` activity, 130
 - for defining rules, 224
 - of `Delay` activity, 130
 - of `EventDriven` activity, 131
 - of `EventHandlingScope` and `EventHandlers` activities, 131
 - of `FaultHandler` activity, 132
 - of `FaultHandlers` activity, 132
 - of `HandleExternalEvent` activity, 133
 - of `IfExists` and `IfExistsBranch` activities, 133
 - of `InvokeWebService` activity, 134
 - of `InvokeWorkflow` activity, 135
 - of `Listen` activity, 135
 - of `Parallel` activity, 135–136
 - of `Policy` activity, 137
 - of `Replicator` activity, 137
 - of `Sequence` activity, 138
 - of `SetState` activity, 138
 - of `State` activity, 138
 - of `StateFinalization` and `StateInitialization` activities, 139
 - of `Suspend` activity, 139
 - of `SynchronizationScope` activity, 140
 - of `Terminate` activity, 141
 - of `Throw` activity, 141
 - of `TransactionScope` activity, 128
 - of `WebServiceFault` activity, 141
 - of `WebServiceInput` activity, 141

syntax (continued)

of WebServiceOutput activity, 142
of While activity, 142

System.* namespaces, contents of, 52–53

system-to-system interaction scenario, 6

T

tabs, adding to Visual Studio Toolbox, 79

TalkWithMeService class, relationship to hosting, 40

/target:assembly option in wcf.exe, description of, 48

/target:codegen option in wcf.exe, description of, 48

/target:exe option in wcf.exe, description of, 48

task completion forms, using, 373–374

Task List in SharePoint, description of, 369

tasks in SharePoint

creating for artist in SharePoint workflow, 385

description of, 366–367

technology leadership, use of Windows Workflow Foundation by, 18

templates. *See* project templates

Terminate activity, using, 141

Terminate method of WorkflowInstance class, description of, 100

TerminationTrackingService sample application, 187–188

ThankYou.aspx for ASP.NET page flow engine, 330

Theme Customization dialog box, accessing in Visual Studio, 86

theme in Visual Studio, availability of, 86–87

threading, defining workflow execution relative to, 43–44

threads, relationship to WorkflowRuntime class, 92–93

Three-state workflow in SharePoint, description of, 366. *See also* state-machine workflows

Throw activity, using, 141, 148, 303

TimeoutDuration property, using with Delay activity, 130

toolbar, options on, 77

ToolboxItem attribute, example of, 153

ToolboxItem class

using with activities, 150

using with workflow designer hosting, 289

ToolboxItemAttribute class

code sample, 79

using in college application communication service, 115–117

trace infrastructure, turning messages on and off in, 309

trace listeners, developing, 309

TraceSwitch class

categorizing messages with, 310

creating instances of, 310

tracing. *See also* debugging workflows

implementing, 308–311

specifying built-in switches for, 311

tracing messages, implementing, 312

tracing options

configuring, 312

for custom applications, 311

tracing switches, using, 311–312

tracking channels, using, 179

tracking database, creating for

SqlTrackingService, 182

tracking profiles

creating, 184

defining, 42–43, 176–177

explanation of, 175–178

modifying, 180–181

serializing, 178

using with SqlTrackingService, 183–184

tracking runtime, explanation of, 174

tracking services, 42–43. *See also* SqlTrackingService class

adding SQL tracking service to workflow runtime, 50–51

developing, 198–202

monitoring workflows with, 174

SqlTrackingService, 181–187

TerminationTrackingService application, 187–188

using, 179

tracking tables, descriptions of, 182–183

TrackingService code sample, 198–200

transaction runtime services, maintaining integrity with, 41

transactional services, relationship to WorkflowEnvironment class, 101

transactions in .NET, resource for, 129

TransactionScope activity, using, 128–129

transitions versus states, 251–252

Translation Management workflow in SharePoint, description of, 365–366

travel reservations, use of back-end transactions in, 127

try-catch block versus faults view, 83

TryUnload method of WorkflowInstance class, description of, 100

Type Browser in Visual Studio, features of, 87–88

Type instance, using with CreateWorkflow method, 94

U

Unload method of WorkflowInstance class, description of, 100

Unsubscribe method, using with e-mail activity, 216

updates. *See* dynamic updates

UpdateTrackingProfile procedure code sample, 184

user data, tracking, 180–181
 using **statement, using with activities, 129**

V

validation, performing in WriteFile activity, 159
variable data types, choosing in SharePoint Designer 2007, 377
 /<vb/cs file list> **option in wcf.exe, description of, 48**
View Code option, using with HelloWorld project, 21–22
view in MVC pattern, description of, 323

Visual Studio

- adding breakpoints in, 305
- advantage of, 71
- Choose Toolbox Items dialog box, 79
- code-beside model in, 64–65
- code-only model in, 65–66
- compiler steps in, 71
- creating solutions and projects in, 72–73
- custom serialization in, 68
- debugging in, 87
- debugging workflows in, 304–306
- defining rules in, 247–248
- design-time validation in, 86
- developing SharePoint workflows in, 383–386
- features of, 47
- markup-only model in, 61–64
- menus in, 75–77
- performing property binding in, 148
- project item types in, 80–81
- projects in, 72
- Properties window in, 81–82
- running web service wizard from, 340
- serialization classes in, 67
- SharePoint workflow in, 385
- solutions in, 71–72
- themes in, 86–87
- Type Browser in, 87–88
- workflow compilation in, 68–70
- workflow serialization in, 66–67
- workflow toolbar in, 77
- WorkflowCompiler class in, 70–71

Visual Studio designers. See workflow designers

Visual Studio development

- installing SharePoint workflow project templates for, 380

- steps required in, 379

Visual Studio Toolbox

- adding items to, 79–80
- adding SharePoint-specific tasks to, 383
- adding tabs to, 79
- displaying custom components in, 78–79
- using with HelloWorld project, 27–30

W

WaitForEmailActivity, **testing execution of, 218–219**

WaitForEmailService **code sample, 213–214**

WaitForIdle **property in ExternalDataEventArgs class, description of, 106**

wca.exe data-exchange service

- command-line switches for, 117
- features of, 48
- generating communication activities with, 117–119

WCF (Windows Communication Foundation)

- ABC's of, 347
- benefit of, 348
- bindings in, 347
- communication contract for, 351
- endpoints in, 347
- exposing workflows as web services with, 349–354
- features of, 59–60, 347–349
- preconfigured bindings in, 348
- role in .NET Framework 3.0, 13
- web resources for, 349

WCF service contracts, defining, 347–348

web references, adding to workflow projects, 337

web service class definitions, creating, 152

web service wizard

- exposing workflows as web services with, 340–343
- limitations of, 342–343
- running from Visual Studio, 340

web services. See also ASP.NET web services

- calling, 336
- calling inside workflows, 335–339
- creating, 335
- exposing workflows as, 340–346, 349–354
- making calls to external web services, 134
- managing sessions with clients, 342
- WCF (Windows Communication Foundation) resources for, 349

Web.config file

- for manual scheduler, 343
- for web service called inside workflow, 341–342
- for WorkflowRuntime class, 322

websites

- BizTalk Server resource, 56
- C# resource, 25
- K2.net, 10
- Lutz's Reflector, 63
- Skelta Workflow.NET, 10–11
- SQL Server Management Studio download, 169

WebServiceFault activity

- description of, 335
- using, 141

WebServiceInput activity

- using, 141–142
- and WebServiceOutput activity, 334

WebServiceInputActivity **class code sample, 152**

WebServiceOutput activity, using, 142

web-services-related activities

- InvokeWebService, 334
- WebServiceFault, 335
- WebServiceInput and WebServiceOutput, 334

wfc.exe compiler

- displaying options for, 69–70
- features of, 48–49
- performing workflow compilation with, 68–70
- using with XAML workflow, 63

While activity

- using, 142
- using activity rules with, 226–227

Windows CardSpace, role in .NET Framework 3.0, 14

Windows Communication Foundation (WCF)

- ABC's of, 347–349
- benefit of, 348
- bindings in, 347
- communication contract for, 351
- endpoints in, 347
- exposing workflows as web services with, 349–354
- features of, 59–60, 347–349
- preconfigured bindings in, 348
- role in .NET Framework 3.0, 13
- web resources for, 349

Windows Event Log, logging workflow instance terminations to, 187–188

Windows Form shopping cart workflow application, 266–270. See also forms; workflow forms

Windows Presentation Foundation (WPF), role in .NET Framework 3.0, 12–13

Windows service, requirement for, 353

Windows Workflow Foundation

- audience for, 16–18
- versus BizTalk Server, 56–57
- deciding on use of, 15–16
- integration with SharePoint, 57–59
- running asynchronously, 43

Windows Workflow Foundation components

- activities, 36–39
- base activity library, 38
- hosting component, 39–40
- rules, 44–46
- runtime services, 40–44
- Visual Studio, 46–47
- workflows, 35–36

Windows Workflow utilities

- wca.exe, 48
- wfc.exe, 48

wizards. See web service wizard

Word 2007, new features in, 356

work batch services. See also batch services

- DefaultWorkflowCommitWorkBatchService, 167
- developing, 190
- features of, 166–167
- SharedConnectionWorkflowCommitWorkBatchService, 167

work batching, relationship to WorkflowEnvironment class, 101

WorkBatch property of WorkflowEnvironment class, using, 167

workflow Actions, defining SharePoint Designer 2007, 378

Workflow Activity Library project template, description of, 75

workflow applications, maintaining integrity in, 41

workflow associations

- making in SharePoint 2007, 368–370
- Start Options for, 369

workflow communication, parameters for, 102–103. See also local communication services

workflow compilation

- performing with wfc.exe, 68–70
- performing with WorkflowCompiler class, 70–71
- steps for, 71

workflow customizations

- debugging in SharePoint, 391–392
- deploying and configuring to SharePoint, 390–391

workflow debugging. See debugging workflows

workflow definitions in filesystem, working with, 292

workflow designer hosting. See also hosts; re-hosting

- description of, 282–283
- event handlers for, 285, 289
- Windows Forms fields and constructor for, 283–284

workflow designers

- activities, 85
- cancellation handler view, 82
- capabilities of, 82
- determining statuses of activities with, 86
- displaying outside bounds of Visual Studio, 278
- faults view, 83
- hosting, 282–283
- location of, 82
- sequential workflows view, 82
- state-machine workflows, 84–85
- workflow view, 82

workflow development styles

- code-beside model, 64–66
- code-only, 65–66
- markup only, 61–64
- markup-only model, 68

workflow engines, features of, 10–11

workflow execution

- process of, 100–101
- using tracking profiles with, 175–178

workflow faults. See faults

workflow forms. See also forms; Windows Form shopping cart workflow application

- association forms, 372
- initiation forms, 373
- modification forms, 374
- task completion forms, 373–374

workflow instance terminations, logging to Windows Event Log, 187–188

workflow instances

- aborting, 172
- ensuring consistency in, 196
- events related to, 96–97
- locking, 173
- obtaining references to, 99
- pausing execution of, 139–140
- setting current states of, 261–262
- starting and managing, 94
- using parameters with, 102–103

workflow loader services

- developing, 202
- features of, 189

Workflow menu, options on, 75–76**Workflow Monitor, using with SqlTrackingService, 186–187****workflow performance counters**

- using, 313–315
- viewing, 314–315

workflow projects, adding web references to, 337**workflow queues**

- creating, 212
- deleting, 212
- role in host-to-workflow communication, 211–212

workflow rules. See rules**workflow runtime. See also runtime services**

- adding SQL tracking service to, 50–51
- code sample, 344
- hosting when using ASP.NET, 320
- managing, 92–93
- managing in ASP.NET, 321–322

workflow scenarios

- application flow, 6
- human-to-system interaction, 6
- system-to-system interaction, 6

workflow serialization, 66–68**workflow stack, 34****workflow tenets, 3–5, 15–16****workflow toolbar, options on, 77****workflow types**

- event-driven workflows, 5, 8
- ordered workflows, 5

workflow view designer, description of, 82**Workflow1 class, modifying for HelloWorld project, 22****Workflow1.cs, 73****Workflow1.designer.cs file, partial class in, 73****WorkflowChanges class, using with dynamic update, 274****WorkflowCompiler class, performing workflow compilation with, 70–71****WorkflowCompleted event handler**

- using with HelloWorld project, 25
- using with web services, 345

WorkflowDesignerLoader class in designer API, using, 278–280**WorkflowEnvironment class, explanation of, 101****WorkflowEventArgs class, using with workflow instances, 96****WorkflowHelper class for page flow engine, 326–328, 329****WorkflowHostService implementation, 352–353****WorkflowIdle event handler, using with communication service, 123****WorkflowInstance class**

- calling Start method of, 100
- methods of, 99–100
- monitoring and manipulating instances with, 99–100

WorkflowInvoked event handler, using with SharePoint workflows, 384**WorkflowMarkupSerializer class, description of, 67****WorkflowMarkupSerializer class, using with activities, 151****WorkflowRuntime class**

- obtaining reference to, 92
- public methods exposed by, 93
- tasks associated with, 92
- Web.config file for, 322

WorkflowRuntime events, debugging workflows with, 306–308**WorkflowRuntime instance, creating, 97–98****workflows. See also SharePoint workflows**

- adding activities to, 160
- adding scheduler services with, 343
- calling web services inside of, 335–339
- creating with SharePoint Designer, 376
- defining in XAML, 62–64
- definition of, 2
- dynamic update from, 275–276
- executing relative to threading, 43–44
- exposing as web services, 142, 340–346, 349–354
- hosting, 91–92
- implementing, 8–11
- indicating dirty workflows, 285
- invoking, 135
- monitoring, 51–52
- monitoring and recording execution of, 42–43
- monitoring with tracking services, 174
- opening for HelloWorld project, 21
- passing first names to, 26
- placing fault handlers in, 301
- retrieving messages from, 25
- sequential workflows, 35–36
- starting in host applications, 40
- statefulness of, 4
- state-machine workflows, 36
- synchronizing execution of, 140
- transparency of, 5
- troubleshooting, 391–392
- viewing .NET code behind, 21–22

WorkflowSchedulerService code sample, 190**WorkflowTrackingLocation, Events property of, 176**

WorkflowType property in ExternalDataEventArgs class

WorkflowType **property in ExternalDataEventArgs class, description of, 106**

WorkflowView **class in designer API, using, 281**

WorkflowwebHostingModule **HTTP module, description of, 342**

workflow.xml file for SharePoint, 381

WorkHandler **property in ExternalDataEventArgs class, description of, 107**

WorkItem **property in ExternalDataEventArgs class, description of, 107**

WPF (Windows Presentation Foundation), role in .NET Framework 3.0, 12–13

WriteFile activity

features of, 152

testing, 159

WriteFileActivity **class code sample, 153–154**

WriteFileActivityDesigner **class implementation, 155–156**

WriteFileActivityDesignerTheme **class code sample, 157**

WriteFileActivityValidator **class code sample, 158–159**

X

XAML (Extensible Application Markup Language)

persisting workflow definitions to, 66–68

relationship to WPF, 12–13

`</XAML file list>` **option in wcf.exe, description of, 48**

XAML workflow, using, 61–64

XML

defining tracking profiles in, 42–43

rules in, 245–247

tracking profile in, 177–178

XML rental rules code sample, 245–246

XML structure for form code sample, 388

XmlWriter **instance, writing for databases, 292**

.xolm file, serializing to, 66–67

XOML files, using with workflow designer hosting, 286

XomlWorkflow.xolm file, **compiling, 70**