

Index

A

- abstraction, 3
- actions
 - completing before processing, 54–55
 - data flowing through, 53
 - passing control between, 51
 - performing, 155–157
 - as round-cornered rectangle, 50
- active classes, 101
- activities, 50, 54–55
- activity diagram window, 51–52
- activity diagrams
 - actions, 50–51
 - activity partitions, 55
 - Browse Catalog use case, 57
 - control flows, 51
 - decision points, 52–53
 - forking and joining flows of control, 54–55
 - icons, 179
 - initial and final nodes, 56–57
 - notation, 180
 - object flows, 53
 - representing page flow, 124
 - Use Case model, 124
- activity group icons, 179
- activity partitions, 55
- actors, 27
 - capabilities provided to, 31–32
 - communicating with use cases, 25
 - creation, 29
 - dialogue with system, 31–32
 - documenting, 30–31
 - ESU Course Registration System, 28–29
 - identifying, 27–28
 - interaction with system, 28
 - naming, 29
 - permissible actions, 27
 - problem statement, 27
 - relationship with use cases, 43–50
 - representation, 27
 - roles, 28
 - use case interaction, 37
 - visualization, 24–25
- Actors overview, 24
- add a student behavior, 70
- Add Diagram > Class Diagram command, 110
- Add Diagram > Sequence Diagram command, 83
- Add Navigation Rule page, 149, 158
- Add Relational Record List wizard, 143–144, 150–151, 154
- Add Session Scope Variable page, 153
- Add UML > Actor command, 29
- Add UML > Attribute command, 87, 111
- Add UML > Class command, 109–110
- Add UML > Operation command, 112
- Add UML > Package command, 103
- Add UML > Partition command, 55
- aggregate design classes, 105
- aggregations, 87–89
- Algebra 101, Section 1 object, 70–71
- Algebra 101, Section 2 object, 70–71
- alternate flow description, 37
- Alternatives combined fragment, 85
- analysis, 8, 10, 65
- Analysis Building Blocks package, 65, 68, 73, 75
- analysis classes, 101
 - changing stereotype, 114–115
 - copying to design package, 114
 - design classes, 105–106
 - design elements, 105–108
 - design models, 105–106
 - design subsystem, 106

- analysis classes, *continued*
 - distributing behavior, 80
 - < < entity > > stereotype, 65
 - ESU Course Registration System, 77–80, 106–108
 - evolution, 101–104
 - functional area packages, 73
 - functional requirements, 105
 - grouping, 65
 - modeling objects, 105
 - significant, 65
 - visualizing for functional area, 78
- analysis model, 63–66
 - boundary classes, 74
 - control classes, 75
 - copying from, 113–114
 - examining relationships, 118
 - operations, 111
 - sequence diagrams, 121
 - use case realizations, 66–68
- analysis model template, 64–65, 73, 83
- annotating classes, 110
- Apache Struts, 135
- APIs (Application Programming Interfaces), 130, 133
- applications, 131
 - displaying data, 144–145
 - list of subjects on the fly, 142–143
 - testing, 159–160
- architecture, xv
- artifact notation, 182
- ASP.NET, 135
- associations, 45, 87, 89–91
- assumptions, validating or rejecting, 17
- attributes, 70, 116
 - classes, 86–87
 - copying, 118
 - creation, 87
 - deleting, 117–118
 - design classes and entity classes, 111
- Attributes section, 120
- auxiliary elements package, 166
- B**
 - basic flow of events, 37–38
 - behavior package, 166
 - behaviors, 111–112
 - Browse Catalog use case realization, 82–85
 - distributing among analysis classes, 80
 - long-lived, 73
 - objects, 70
 - performance in workflow, 50–51
 - Billing System actor, 29, 31, 34
 - Blank Model template, 100
 - Booch, Grady, xv–xvi, 5–6, 72
 - boundary classes, 72–74, 120–122
 - Analysis Model, 74
 - entity and control classes, 103
 - finding, 77
 - packages, 103
 - < < boundary > > stereotype, 65
 - Browse by Category sequence diagram, 85
 - BROWSE BY SUBJECT AREA event, 38
 - Browse by Subject Area function, 38
 - BROWSE COURSE CATALOG event, 38
 - Browse Course Catalog Participants classes diagram, 90
 - Browse Course Catalog Participants diagram, 106
 - Browse Course Catalog sequence diagram, 121
 - Browse Course Catalog scenario, 121
 - Browse Course Catalog use case, 35, 38–40, 42, 48, 57, 131, 137–138
 - Browse Course Catalog use case realization, 77–78, 82–85, 90
 - browse package, 122
 - BrowseCatalogHome boundary class, 77, 90, 121
 - browsecataloghome page, 131, 138, 140
 - browsecataloghome.jsp file, 142–143, 145, 147
 - browsecataloghome.jsp page, 150, 152, 159–160

- browsecatalogscreen page, 142–143
 business modeling, 8, 10
- C**
- CANNOT RETRIEVE COURSE OFFERING INFORMATION event, 39
- case model template, 24–25
 case post-conditions, 37
 case pre-conditions, 37
 class diagrams, 65, 109
 - association adornments, 172
 - changing stereotype relationship, 115
 - classes, 90
 - as collection of model elements, 109
 - copying from analysis, 113
 - deleting extra elements, 118
 - design classes, 110–112
 - design element information, 113
 - icons, 171
 - metamodel, 165
 - model, 113–114
 - organizing into packages, 110
 - quickly adding elements, 112
 - relationship, 119–120
 - visualizing relationship, 116–118
- classes, 102
 - active, 101
 - annotating, 110
 - attributes, 86–87
 - behaviors, 72, 102, 107
 - bidirectional semantic connection
 - between, 87
 - boundary classes, 74
 - << boundary >> stereotype, 65
 - breaking up, 76
 - class diagram, 78, 90
 - combining, 76
 - as compartmentalized rectangles, 72
 - contents, 172
 - control classes, 74–75
 - << control >> stereotype, 65
 - creation, 75
 - defining, 70–72
 - deleting, 108
 - determining correct abstractions, 76
 - distinguishing from objects, 71–72
 - documenting, 75–76
 - dynamic behavior, 99
 - entity classes, 73
 - fine-grained responsibilities, 101
 - functionally related, 102
 - good candidate for, 76
 - grouping, 102, 107
 - incohesive, 86
 - instances, 71
 - major theme, 71
 - naming, 71–72, 76
 - purpose, 75
 - relationships, 87–91
 - structure, 72, 107
 - types, 72–75
 - unnecessary, 108
 - view of participating, 87–91
- classes package, 166
- closed state, 70
- code, throw-away, 17
- collaboration, 174
- collaboration diagram, 171
- Column Selection and Other Tasks page, 151, 154
- combined fragments, 85
- com.esu.registration.<package_name>, 101
- Command-hyperlink component, 147–148
- commandExRowAction, 157
- commandLink component, 152
- common behavior package, 166
- common semantics, 71
- communicate association, 45
- << communicate >> stereotype, 49
- communication, 23
- communication diagram notation, 182
- complex systems models, 3
- complexity and sequence diagrams, 82
- component diagrams, 99, 171
- component specification package, 99
- components, 99, 102

186 Index

- composite aggregation, 89
 - composition relationships, 89, 91, 118
 - concurrent activities, 54–55
 - conditional logic, 82
 - conditional structures modeling, 85
 - Conditions page, 151, 154
 - configuration management, 102
 - Configure Data Controls page, 145
 - Connection wizard, 144
 - connectors, 173
 - constraints, 165
 - Construction phase, 10, 12
 - Context Overview diagram, 25
 - control classes, 72–75, 77
 - control flows, 51
 - < <control> > stereotype, 65
 - Copy command, 113
 - copying attributes and operations, 118
 - Core JavaServer Faces* (Geary and Horstmann), 137
 - Course class, 86, 90, 115–116, 120
 - Course Data Objects package, 108
 - Course design class, 116
 - Course entity class, 78, 106–108
 - Course object, 86
 - Course Offering Name detail, 39
 - Course Offering Number detail, 39
 - Course Offerings label, 147
 - Course Registration analysis package, 108
 - Course Registration functional area, 25–26, 66
 - Course Registration package, 29, 122
 - COURSE table, 151
 - CourseCatalog control class, 77, 90
 - COURSEID, 157
 - courseID variable, 153
 - CourseList boundary class, 77, 90, 121
 - courselist data object, 151
 - CourseOffering analysis class, 111, 116, 118
 - courseoffering attribute, 120
 - CourseOffering boundary class, 71, 122
 - CourseOffering class, 72, 86–87, 115, 120
 - courseoffering data object, 154
 - CourseOffering entity class, 78, 106–108
 - courseoffering record list, 154
 - CourseOfferingDetail boundary class, 77, 121–122
 - CourseOfferingList boundary class, 77, 121
 - courseofferings page, 131, 138, 141
 - courseofferingsdetail.jsp page, 152–153, 158
 - courseofferings.jsp page, 142, 149, 156
 - data table, 152
 - filtering or limiting courses displayed, 150–152
 - navigation rule, 157–158
 - passing parameters, 147–148, 154–157
 - COURSEOFFERING table, 154
 - Create course catalog use case, 35
 - Create New Composition Association command, 119
 - Create New Datastore Node option, 53
 - creativity, 8
 - custom JSF components, 143
 - CWM (Common Warehouse Metadata), 166
- D**
- data
 - connection creation, 143–144
 - flowing through actions, 53
 - managing, 131
 - data beans, 131
 - data flow, 53, 87
 - data object creation, 143–144, 150–151, 153–154
 - data source mediators, 132–133
 - data sources, 131–132
 - data stores, 53, 132
 - data table, 144–145, 147–149, 152, 155–156
 - data types, 132, 166
 - databases, 19, 143–144
 - dataTable properties, 147
 - Day(s) detail, 39
 - DDL (Data Definition Language) statements, 129
 - decision points, 52–53

- decisions
 - capturing and documenting, 12
 - naming, 52
 - Define a New Server page, 159
 - delete a student behavior, 70
 - Delete from Diagram command, 118
 - Delete from Model command, 118
 - department attribute, 86
 - Department class, 87
 - deployment, 8, 11
 - deployment diagrams, 99, 171
 - design, xv, 8–9, 120
 - design class diagram, 110–111
 - design classes, 105–106
 - adding to class diagram, 110–111
 - attributes, 111
 - boundary classes, 121
 - manually adding, 109
 - operations, 111–112
 - design contract package, 99
 - design elements, 98, 101–104
 - analysis classes, 101, 105–108
 - class diagram, 113
 - Design Model, 97–100, 103–105
 - boundary classes, 122
 - persistence agnostic, 129
 - Design Model Class diagram, 118
 - Design Model diagram, 104
 - Design Model file, 100
 - design model relationship, 120
 - Design Overview diagram, 104
 - design packages, 102–103
 - copying analysis class to, 114
 - implementation model version, 129
 - design subsystems, 102, 106
 - design-level use case realizations, 99
 - detailed design model, 98
 - developerWorks Web site, xix
 - development
 - Rational Software Architect, 12–13
 - as series of iterations, 8
 - development life cycle, 8–9
 - guideline, 12
 - phases and iterations, 10
 - development phases, 11–12
 - development team, 102
 - diagram.gph file, 152, 159
 - diagrams
 - adding classes, 78
 - standardizing, 6–7
 - discipline, 8
 - DISPLAY COURSE OFFERINGS event, 39
 - Display course offerings links, 149, 159
 - displaydetails, 158
 - documenting
 - actors, 30–31
 - classes, 75–76
 - use cases, 35
 - Domain Model diagram, 65
 - Draw > Draw All command, 160
 - dynamic semantics, 165
 - Dynamic Web Project, 138
 - Dynamic Web Projects >
 - ESUCourseRegistration >
 - Web Diagram command, 139
- E**
- EJB (Enterprise Java Bean), 135
 - EJB (Enterprise Java Bean) Entity Beans, 131
 - Elaboration phase, 10, 12
 - use case model, 23
 - use case specification, 37
 - elements
 - class diagram as collection, 109
 - deleting, 117–118
 - extended, 48–49
 - lifelines, 83–84
 - enterprise information systems, 132
 - entities, representing, 68–70
 - Entity classes, 72–73
 - attributes, 111
 - copying, 113
 - finding, 77–78
 - persistent, 106
 - <<Entity>> label, 118
 - <<entity>> stereotype, 65

- Establish a connection to a database
 - page, 144
 - ESU Course Registration System
 - actors, 28–31
 - analysis classes, 77–80, 106–108
 - assigning professors to courses, 18–19
 - background, 18–19
 - boundary classes, 77
 - Browse Course Catalog scenario, 124
 - control classes, 77
 - Course Registration functional area,
 - 25–26, 66
 - database layout, 129
 - design model, 100
 - entity classes, 77–78
 - initial design of user interface, 121–122
 - problem statement, 19–20
 - registering students, 18–19
 - risks, 19
 - storing information in relational database, 129
 - System Maintenance functional area,
 - 25–26, 66
 - use case diagrams, 45–47
 - use cases, 34–35
 - Web application frameworks, 134–136
 - Web-based user interface, 134–136
 - ESUCourseRegistration dynamic Web
 - project, 138
 - ESUCourseRegistrationEAR Enterprise
 - Application, 138
 - exceptional flows, 37
 - extend relationship, 48–49
 - extension points, 37
 - external interfaces, identifying, 17
- F**
- Faces Components folder, 148
 - File > Run > Other command, 138
 - Filters page, 151, 154
 - final node, 56–57
 - Find/Replace... command, 26, 35, 66, 68
 - flow of control, 52, 54–55
 - flow of events, 73
 - fork node, 54–55
 - forms, 120, 122
 - frameworks, managing data, 131
 - < functional area > Analysis Elements package, 75
 - `{functional.area}` package, 25–26, 65–66
 - functional area packages, 25–26, 35, 66, 73
 - functional areas, 78
- G**
- generalization, 173
 - "Get offering information" operation, 113
 - grouping, 102
 - analysis classes, 65
 - functionally related classes, 107
 - use cases, 25
 - guard conditions, 52–53
 - GUI tools, 132
- H**
- h:commandLink, 148
 - h:dataTable, 147
 - high-level design model, 98
 - high-ceremony process, xv
 - HTML, 137
- I**
- IBM DB2 Universal Database, 129, 143
 - IBM Rational Requisite Pro, 40–42
 - IBM Rational Software Architect, 12–13, 158
 - actors, 27
 - analysis model creation, 63–64
 - Analysis Model template, 64–65, 73
 - case model template, 24–25
 - data modeling capabilities, 129
 - JSF development, 136–138
 - Struts-based Web applications, 135
 - transforming between models, 118
 - visualizing code in UML format, 129
 - Web Development capabilities, 138
 - IBM Rational Unified Process, 9–12, 72
 - IBM Web site, 13

- ID parameter, 148
 - ideas, solicitation, 17
 - if, then, else logic, 82
 - implementation, 8, 10, 97, 129–131
 - implementation model, 97
 - JSF (Java Server Faces), 136–137
 - reducing complexity, 129
 - SDO (Service Data Objects), 131–133
 - user interface choices, 134–136
 - implementation subsystems, 129
 - Inception phase, 10–12, 17–20, 23
 - include relationships, 48–49
 - information
 - long-lived, 73
 - persistence, 129–130
 - storing and accessing, 19
 - infrastructure, 166
 - initial node, 56
 - Insert Record List wizard, 145
 - instances, 71
 - interacting with system, 27
 - internal structure, 173
 - internal wiring, 173
 - inverted Internet domain naming convention, 101
 - iterations, 8–9, 12
- J**
- J2EE platform, 100, 131
 - The J2EE™ Tutorial, Second Edition* (Bodoff et al.), 137
 - Jacobson, Ivar, xv, 5–6
 - Java
 - code, 157
 - inverted Internet domain naming convention, 101
 - java.lang.String variable type, 153
 - JavaServer Faces, 135
 - JDBC (Java Database Connectivity), 131, 135
 - join node, 54–55
 - JSF (Java Server Faces), 135–137
 - JSP, 135
 - JSP navigation map, 157
 - JSTL (JSP Standard Tag Library), 135
- K**
- Kernel, 166
 - Key Abstractions diagram, 65
 - Key Controllers diagram, 65
 - keyvalue, 157
- L**
- layered design model, 98
 - lifelines, 83–84
 - lifetime dependency, 89
 - line-oriented interfaces, 103
 - link page, 148–150
 - location attribute, 86
 - Location detail, 39
 - logical design elements, 102
 - Loops combined fragment, 85
- M**
- Main diagram, 104, 122
 - main use case diagrams, 44
 - Maintain course information use case, 35
 - Maintain student information use case, 35
 - major functionality, identifying, 17
 - messages
 - creation, 84
 - notation, 182
 - UML notation, 81
 - metacircular interpreter, 165
 - metadata, 133
 - metamodel, 165–168
 - metamodel classes, 165
 - method wars, 6
 - Model Driven Architecture initiative, 63
 - Model Explorer, 23, 29, 31, 40, 44–45, 49, 55, 63, 75, 83, 87, 90, 100, 103
 - modeling and conditional structures, 85
 - models, 3–5
 - copying elements, 25
 - static structure, 109–120

model-view-controller point of view, 72
MOF (Meta-Object Specification), 166

N

namespace mapping, 122
navigation diagram, 124
navigation rule, 157–158
New > Relational Record List command,
143, 150, 153
New > Scripting Variable > Session
Scope Variable command, 153
New > UML Model command, 23, 63,
100
New Database Connection wizard, 144
nodes, 139–140, 182
notations, 4–6
numberOfferingsInDepartment
attribute, 86

O

object flows, 53
objects, 68–71, 81
adding to sequence diagram, 83–84
distinguishing from classes, 71–72
structure, 86–87
value for every attribute, 86
offerNumber attribute, 86
OMG (Object Management Group), 7,
63, 165
OMG web site, 7, 165
OMT, 5–6
OMT-2, 6
OOSE, 5–6
open state, 70
operations, 111–112, 118
Overview package, 24–25, 65
owning
package, 83
project, 23

P

packages
boundary classes, 103

class diagram, 110
design elements of system, 98
Design Model diagram visual surface,
104
functionally related classes, 102, 107
names valid in implementation plat-
form, 122
notation, 175
organizing design model, 98
sequence diagrams, 99
structure, 122–123
subpackages, 98
visibility into, 108
page classes and boundary classes, 120, 122
Palette, 56–57
Association icon, 45, 90
Composition Association icon, 91
Lifeline icon, 83
Synchronous Message icon, 84
Text option, 104
parent-child relationships, 90
parts, 173
passwords and databases, 144
Paste command, 114
*Patterns of Enterprise Application Architec-
ture* (Fowler), 131
persistence choices, 129–131
platform-neutral models, 129
ports, 173
problem statement, 19–20, 27
process components dimension, 10–12
processes, 4, 8–9
Professor actor, 28–29, 31, 34
Professor and Prerequisite Courses
detail, 39
profiles package, 168
programming model and JSF (Java Server
Faces), 137
projects, 17–20
"proof of concept" prototypes, 17
properties, 70
Properties view, 115–116, 120, 147–148,
156–157

Properties window, 76, 83
prototypical use case realization, 65
prototypical use cases, 25
prototyping user interface, 74, 121

Q

QuickEdit view, 156, 157
QUIT event, 39

R

Rational Software Modeler, 27
realized nodes, 139–140
Record List Properties page, 143, 151, 154
Refactor > Rename command, 104
Register for courses use case, 35, 48
Registrar actor, 29, 31, 34
related things, grouping, 102
relational databases, 129, 132
Relational Record List page, 143, 150, 154
Relational Record List wizard, 143, 150, 153
relationships, 87–91
 adding, 119–120
 analysis classes becoming, 106
 stereotype changing, 115
 visualizing, 116–118
Replace >> button, 26, 66
Request course roster use case, 35
request scope variable, 153
Requirement Explorer, 40, 42–43
requirements, 8–9, 10
RequisitePro project, 40
responsibilities, 101–102
result set, 132
right projects, 17–18
risk-mitigating processes, 8–9
risks, 8–9, 17, 19
row action, 155–157, 160
Ruby on Rails, 135
Rumbaugh, Jim, xv, 5–6
Run on Server... command, 159
RUPAnalysis::Entity stereotype, 118

S

screen, search criteria, 121
script variable, 157
scripting variables, 152–157
SDO (Service Data Objects), 131–133, 137, 143
SDO datagraph, 132
search criteria, 142–143
SEARCH FOR A COURSE OFFERING event, 39
Search for a Course Offering or Quit function, 38
Select a Database page, 144
Select courses to teach use case, 35
Select Page Data Object page, 148, 154
Select Requirement in > RequisitePro command, 41
SELECT SUBJECT AREA event, 39
semantic models, standardizing, 6–7
sequence diagrams, 25, 65, 80–84
 analysis model, 121
 Browse Course Catalog use case realization, 90
 combined fragments, 85
 notation, 181
 packages, 99
 use case realization, 87
service data object, 144–145
significant analysis classes, 65
sketch and code approach, 97
software architecture, xv
software development, 9–12
software systems, 3
source class, 119
specification document structure, 165–166
Specify connection parameters page, 144
state machine diagrams, 99, 177
state machine notation, 178
stereotyped component, 98
stereotypes, 48–49, 114–115
storyboarding, 74
structure and objects, 86–87

- structure package, 166
 - Struts, 135
 - Student actor, 28–30, 34
 - Student class, 71, 75–76
 - StudentHistory class, 71
 - SUBJECT table, 144
 - SUBJECTID, 148
 - subjectlist service data object, 145
 - subjectlist > subjectlist, 148
 - subjectlist table, 145
 - submit relationship, 120
 - subsystem specification model, 98
 - subsystems, 98–99, 101–102
 - superstructure, 166
 - syntactic notation, standardizing, 6–7
 - system architecture, xv
 - system behavior, 23–26
 - system interfaces, 74
 - System Maintenance analysis package, 108
 - System Maintenance functional area, 25–26, 66
 - System Maintenance package, 29
 - systems, 31–32, 65–66, 98–99
 - actor interaction, 28
 - flow of events, 37
 - functional requirements, 23
 - interacting with, 27
 - overview, 24
 - performing functionality, 66–68
 - representing threads of control, 101
 - surroundings-dependent part, 74
- T**
- target class, 119
 - target implementation platform, 122
 - templates, 71, 175
 - testing, 8, 11–12, 159–160
 - throw-away code, 17
 - Time detail, 39
 - time dimension, 10–12
 - time-based phases, 10–12
 - timeOfDay attribute, 86
 - tools, 4, 131
 - Transition Phase, 10
 - triangle for success, 4
- U**
- UI diagram, 65
 - UML (Unified Modeling Language), 5–7, 165–166
 - ability to modify, 168
 - actor represented as stickman, 27
 - capturing and documenting decisions, 12
 - classes in target classes with appropriate changes, 118
 - subsystems, 98–99
 - UML 2.0, 7
 - UML 2.0 Superstructure, 165–166
 - UML semantic model structure, 165
 - units of control, 102
 - unrealized nodes, 139–140
 - Use-Case Building Blocks package, 25
 - use case diagrams, 43–50, 176
 - {use.case} element, 65
 - use case model, 23–25
 - use case model activity diagram, 124
 - Use Case Model option, 23
 - #{use.case} package, 68
 - use case realization, 66–68
 - analysis classes, 80
 - classes and relationships, 87–91
 - design-level, 99
 - prototypical, 65
 - sequence diagrams, 87
 - use case specification, 36
 - Browse Catalog use case, 38–40
 - document, 41–42
 - Elaboration phase, 37
 - looking at nouns in, 77–78
 - standard template for creation, 37–38
 - use cases, 31–32
 - actors, 25, 33, 37, 43–50
 - brief description, 36
 - conditional behavior, 48
 - creation, 35
 - data, 37
 - data flow, 50–57
 - database requirements, 40–41
 - definition, 32
 - documenting, 35

- ESU Course Registration System, 34–35
 - exceptional flows, 37
 - extend relationship, 48–49
 - extending, 37
 - flow of control, 50–57
 - flow of events, 73, 75
 - functional area package, 35
 - functional decomposition, 33
 - functionality, 33–34, 36
 - goodness, 33–34
 - grouping, 25
 - identifying, 32
 - include relationships, 48–49
 - level of detail, 33
 - limiting number, 34
 - managing sequence of messages, 65
 - most important, 25
 - naming, 35
 - normal sequence of events, 37
 - optional activity, 25
 - optional behavior, 48
 - prototypical, 25
 - purpose, 36
 - relationships between, 43–50
 - represented as oval, 32
 - running, 74
 - sequence diagrams, 25, 80
 - sequencing behavior specific to, 74–75
 - starting and ending, 37
 - text explaining purpose, 35
 - use case realizations, 67
 - verification of user, 48
 - what system should do, 66
 - Use Cases package, 40
 - User Interface diagram, 122
 - user interfaces, 74, 99, 101–102, 120, 173
 - choices, 134–136
 - design, 65, 121
 - dynamic Web applications, 136–137
 - exposing to end users, 121
 - initial design, 121–122
 - package structure, 122–125
 - prototyping, 121
 - realization, 174
 - self-contained rich client, 134
 - separating from system, 134
 - Web-based, 134
 - User Verification use case, 48
 - userid, 144
- V**
- variable row, 157
 - Versatile Actors package, 25, 29
 - view of participating classes diagram, 87–91
 - visual modeling, xvii–xviii, 3
 - visual summary of notations, 169–182
 - visualization, xv
 - Visualize > Add to Current Diagram, 116
 - visualizing relationship, 116–118
- W**
- Web application diagram, 138
 - Web application frameworks, 134–136
 - Web applications, 136, 138
 - Web diagram, 138–142, 152, 159–160
 - Web diagram editor, 138–139
 - Web page editor, 143, 150, 153
 - Web pages, 140–143, 146, 148–150
 - Web Perspective, 143
 - Web project, 138
 - Web-based user interface, 134–136
 - WebSphere Application Server V6, 158
 - Window > Show View > Properties command, 31
 - Window > Show View > Requirement Explorer command, 40
 - window-oriented interfaces, 103
 - work product, 102
 - workflow, 50–52, 56–57
- X**
- XML configuration files, 138–139