

# INDEX

## Numbers

- 3 A's, 106
- 7-zip archives, 504–507, 514

## A

- accessing
  - databases, 576
  - nonpublic members from tests, 122–126
  - PROD, 465–467
  - records in data-driven Web performance tests, 293–294
  - rules in code analysis, 155, 162
  - Team Foundation Server. *See* accessing Team Foundation Server
  - Team Web Access for. *See* Team Web Access
- accessing Team Foundation Server
  - from Eclipse, 398–399
  - introduction to, 392
  - new features in TFS 2010 for, 400
  - through Web browsers, 395
  - from Visual Studio, 392–394
- action logs, 262
- action recordings
  - for coded user interface testing, 351–354
  - defined, 262
  - Test Runner for, 333–335
- Active mappings, 427
- Active work items, 556–559, 626
- activity diagrams
  - adding to use case diagrams, 27
  - concurrent flow in, 23
  - creating, 26–27
  - data flows in, 23–25
  - introduction to, 22
  - in software architecture, 11–12
  - toolboxes in, 25–26
  - in top-down design, 22–27
  - understanding, 22–23
- actors, 18–21, 28
- Administration Console, 394
- administrators
  - access in code analysis, 162
  - of databases, 206
  - in Source Control Explorer, 428
  - in Team Foundation Server, 402, 415
  - in Team Foundation version control, 426
  - of Web sites, 279
- advanced features
  - for branching plan implementation, 469
  - for database unit testing, 230–231
  - in Intellitrace, 245–246
  - in Performance Explorer, 181–182, 184
- agents
  - in build process, 495–496
  - defined, 278
  - in distributed load testing, 313–315
  - in Lab Management, 359
- aggregating data, 574
- Agile Estimation and Planning, 608
- Agile Management for Software Engineering*, 565
- Agile Manifesto, 551
- agile planning workbooks. *See also* MSF for Agile Software Development v5.0
  - capacity planning worksheets in, generally, 612
  - interruptions worksheets in, 607
  - introduction to, 599–600
  - issues spreadsheets in, 613
  - iteration backlog workbooks in, 608–613
  - iteration planning with, 607–608
  - iterations worksheets in, 606–607
  - product backlogs in, 600–601
  - product planning workbooks, 602–607
  - release planning in, 600–601
  - retrospectives in, 613
  - team capacity worksheets in, 612
  - team member capacity worksheets in, 613
  - tracking iterations in, 613–614
  - worksheets for product backlog, 603–606
- Agile Software Development. *See* MSF for Agile Software Development v5.0
- alerts, 492
- All Links tabs, 557
- Allocation views, 193–194
- Analysis options, 66–68
- Analyze, 171–173
- analyzing applications. *See* Architecture Explorer
- analyzing code. *See* code analysis tools
- analyzing test impact. *See* test impact analysis
- Anderson, David J., 565
- APIs (application programming interfaces), 407

application programming interfaces (APIs), 407

application tier

- in build architecture, 477
- in hosted environments, 419
- in multi-server installations, 412
- in Team Foundation Server, 380–381, 407–411
- in three-tier architecture, 405

application time, 187

applications analysis. *See* Architecture Explorer

architecture

- CMMI Level 3 in, 565
- layer diagrams in, 84
- process templates in, 565
- of Team Foundation Build 2010, 477–478

Architecture Explorer

- classes in, 66–68
- code base of, 60
- dependency graphs, creating
  - first, 71–73
- dependency graphs, generally, 71
- dependency graphs in, 71–73
- dependency graphs, legends in, 77–78
- dependency graphs, navigating through, 74–77
- dependency graphs, toolbars in, 78–80
- introduction to, 14–15, 59–61
- layer diagrams in, 84
- members in, 68–69
- namespaces in, 64–66
- navigating in, 62–64
- options in, 62–69
- queries in, 69–71
- windows in, 61–62

archives, 510

Areas and Iterations, 528, 627

Arrange, Act, and Assert, 106

artifacts, 85–86, 433

ASP.NET Development Server, 280–282

ASP.NET Profiler, 262

ASP.NET Web applications, 278–279

ASPX pages, 177

assemblies, 8, 74–76

AssemblyCleanup attribute, 112–113

AssemblyInitialize attribute, 112–113

assembly-level suppression, 151

Assert.AreEqual, 113–114

Assert.AreNotEqual, 113–114

Assert.AreNotSame, 114–115

Assert.AreSame, 114–115

Assert.Fail, 116

Assert.Inconclusive, 116

assertions

- in action recordings, 353–354
- in coded user interface tests, 342–345
- in unit testing, 112–116

Assert.IsFalse, 115

Assert.IsInstanceOfType, 115–116

Assert.IsNotInstanceOfType, 115–116

Assert.IsNotNull, 115

Assert.IsNull, 115

Assert.IsTrue, 115

associate check-ins, 389

associations, 47–48, 53–54

asynchronous messages, 29–31

atomic check-ins, 389, 424

Attachments tabs, 557

attributes

- AssemblyCleanup, 112–113
- AssemblyInitialize, 112–113
- of class diagrams, 47–48, 51–57
- ClassCleanup, 112
- ClassInitialize, 112
- defined, 5
- in distributed load testing, 315
- TestCleanup, 111
- TestInitialize, 111
- TestProperty, 119–120

authoring test cases, 327–329

automated software testing. *See also* software testing

- analyzing, 273–274
- categories of, 267
- creating, 265–266, 272
- executing, 273–274
- exporting, 270
- filtering, 270

- grouping, 270
- introduction to, 265
- options in, 267
- ordered tests, 272–274
- properties of, 273
- publishing, 271
- sorting, 270
- test impact analysis in, 275–276
- test projects, 265–267
- Test Results for, 269–271
- test settings in, 274–275
- in Test View, 267–269
- in XML or TRX files, 270

automated tests

- in Lab Management, 371–375
- manual, 336–338
- of software. *See* automated software testing
- in Team Foundation Build 2010, 497–500

automation. *See* Team Foundation Build 2010

Automation Status, 352

AutoMerge, 465

## B

BaseIntrospectionRule, 156–157

baseless merging, 449. *See also* merging

baseline reports, 186

BaseStaticAnalysisRule.cs, 157

Beck, Kent, 122, 600

bi-directional dependencies, 87–88

binaries, 504–508

Binaries property pages, 184

Binary Archive files, 510

binding to sources, 294

black box testing, 100

blank performance sessions, 172–173

“Block incremental deployment if data loss might occur”, 223

blocking test cases, 331

branching, 389. *See also* merging

- advanced plans for, 467–469
- avoiding, 449–450
- code promotion in, 451
- defined, 449

- definitions in, 449
- folders in, 421
- implementing plans for.
  - See* branching plan implementation
- introduction to, 423–424, 447–448
- objects in, 442
- per feature, 451–452
- per release, 450–451
- plans for, generally, 452
- scenarios in, 453–454, 468
- in Software Configuration Management, 448
- strategies for, 449–452
- in Team Foundation version control, 440–443
- understanding, 448–449
- branching plan implementation
  - advanced, 469
  - Branching Visualization in, 467
  - Branching Wizard for, 457
  - bugs in, 461–466
  - conflicts in, 461–465
  - creating new team projects, 454
  - identifying branches in, 455–456
  - Merge Tool for, 463
  - Merging Wizard for, 458–461
  - Pending Changes window in, 460–463
  - permissions in, 466–467
  - read-only settings in, 466–467
  - Solution to Version Control in, 454–455
  - Source Control Explorer in, 465–466
  - specifying branch owners in, 456
  - Branching Visualization, 467
  - Branching Wizard, 457–458
  - breaking the build, 155
  - breakpoints, 109–110, 251–252
  - browsers
    - in load testing, 303
    - Performance Explorer in, 186
    - in Web performance tests, 286–287
  - buddy builds, 475, 491
  - bug work items, 557–559
  - bugs. *See also* debugging
    - in branching plan implementation, 461–466
    - reports on, 561, 567
    - on Test Runner, 336, 369–371
  - build agents
    - defined, 474
    - introduction to, 475–478
    - in virtual environments, 359
    - ZIP archives on, 504
  - Build Defaults, 486–487
  - build definitions
    - creating, 482–489
    - defined, 478
    - in team projects, 391
  - Build Explorer, 479–481
  - build masters, 483. *See also* owners
  - Build option in Source Control Settings, 435
  - Build plug-ins, 618
  - build process templates
    - in automated build-deploy-test, 372–374
    - re-using, 509–512
    - workflow activities in, 513
  - build verification tests (BVTs), 268–269
  - build-deploy-test with virtual environments, 371–375
  - builds
    - agents in. *See* build agents
    - artifacts in, 433
    - code analysis tools in, 155
    - controllers in, 475–478
    - definitions in. *See* build definitions
    - deletion options in, 477
    - details in, 478, 481–482
    - layer diagrams in, 90–91
    - in manual testing, 323–324
    - notifications in, 476, 491–492
    - numbers of, 497
    - output of, 206–207
    - parameters in, 494–500
    - process templates in. *See* build process templates
    - quality of, 480–481
    - reports on, 561, 567
    - status notification for, 473
    - Team Foundation Build 2010 for. *See* Team Foundation Build 2010
    - in team projects, 383, 386
    - in test plans, 323–324
  - Builds node, 527
  - burn rate reports
    - in agile planning workbooks, 560–561
    - burn-down and, 592–595
    - in CMMI Level 3, 567
  - burn-down
    - burn rate and, 592–595
    - charts of team progress, 613
    - dashboards, 593–594
    - reports in agile planning workbooks, 560–561
    - reports in CMMI Level 3, 567
    - worksheets in iteration backlog workbooks, 609
  - business analysts, 564
  - Business Intelligence Development Studio, 587
  - Business Logic Layer, 84–85, 87–89
  - Butterfly mode, 79–80
  - BVTs (built verification tests), 268–269

## C

- C# Console Application, 170
- CA1709, 150
- CAL (Client Access License), 392
- Call Attributed Provider (CAP), 169
- Call Tree view, 190, 193
- Caller/Callee view, 192–193
- calling to databases, 233
- calls history, 250
- CAP (Call Attributed Provider), 169
- Capabilities tabs, 363–364
- Capability Maturity Model
  - Integration (CMMI). *See* MSF for CMMI Process Improvement v5.0
- capacity planning worksheets, 609, 612
- Carnegie Mellon Software Engineering Institute, 562
- change management, 202–203
- changesets
  - in check-ins, 433
  - in History windows, 437

- changesets (*continued*)
  - in impact analysis, 324
  - introduction to, 423–424
  - merging, 459
  - rollbacks to, 445
  - tracking, 443–444
- channels, 434
- Check Constraint script, 220–221
- check-in policies
  - for cross-platform and Eclipse clients, 399
  - in Team Foundation Server, 390
  - in Team Foundation version control, 423–424, 435–436
- checking in
  - defined, 432
  - introduction to, 432
  - items, 433–434
  - merged changes, 460–461
  - notes for, 390, 434
  - policies for. *See* check-in policies
- checking out, 432–435
- cherry-picked merges, 456
- child folders, 442–443
- CI (continuous integration) triggers, 391, 484
- Circular Reference analyzer, 76
- circular references, 66
- Class Coupling, 163–164
- class diagrams
  - association properties of, 53–55
  - attribute properties of, 51–52
  - creating, 55–57
  - introduction to, 46–47
  - operations properties of, 52–53
  - in software architecture, 13
  - toolboxes in, 48–49
  - type properties of, 49–50
  - understanding, 47–48
- ClassCleanup attribute, 112
- classes
  - defined, 47
  - diagrams of. *See* class diagrams
  - options for, 66–68
  - in profiling applications, 170–171
  - of unit tests in Visual Studio Express Edition, 104
- Classification plug-ins, 618
- classifications, 383, 386
- ClassInitialize attribute, 112
- Clean Workspace, 496
- cleanliness of databases, 232
- cleanup, 110–113
- Clickadd() method, 353
- clicking, 127
- Client Access License (CAL), 392
- clients, 313, 405
- cloaking folders, 427
- clock cycles, 184
- Closed user story work items, 556
- Closed work items, 557–560, 626
- CMMI (Capability Maturity Model Integration). *See* MSF for CMMI Process Improvement v5.0
- code analysis tools
  - for ASP.NET applications, 144
  - base rules in, 156–157
  - command-line, 151–155
  - correcting problems in, 148–149
  - deploying rules in, 162–163
  - enabling, 144–146
  - executing, 146–147
  - existing rules in, 163
  - FxCop and, 140–141
  - FxCopCmd options in, 151–154
  - FxCopCmd project files in, 154–155
  - IIIntrospectionRule interface in, 159
  - implementing rules in, 157–161
  - integrating build process, 155
  - introduction to, 139–140
  - introspection in, 155–156
  - Microsoft.Cci assembly in, 157–158
  - need for, 140
  - new rules, creating, 156–157
  - reflection in, 155–156
  - rule violations in, 147–151
  - rules, built into, 142–143
  - rules, creating, 155–157
  - rules sets in, 97, 143–144
  - Rules.xml in, 161–162
  - in software development, 97
  - in Source Control Settings, 435
  - suppressing messages in, 149–151
  - writing rule implementation code in, 159–161
- code base, 60
- code coverage
  - in build process, 498–500
  - data adaptor, 262–263
  - enabling, 129–131
  - in Unit Test Framework, 100, 110
- code generation, 126–129
- Code Metrics tool, 139, 163–164
- code promotion, 451
- Coded UI Test Builder, 340–345
- coded user interface (UI) testing
  - action recordings for, 351–354
  - adding tests in, 341–342
  - Builder for, 340–345
  - data-driven tests in, 347–349
  - data-driven Web performance tests vs., 292
  - enhanced assertion reporting in, 350–351
  - generated code in, 345–346
  - introduction to, 339–340
  - running tests in, 347
  - sample applications in, 340–341
  - of software, 261
  - supporting technologies and, 354
  - test projects, creating, 341
  - using() clause in, 349–350
  - Web performance tests vs., 278–279
- coded Web performance testing, 281, 294–297
- Cohn, Mike, 555, 608
- CollectionAssert class, 116–118
- command-line
  - build tools at, 490
  - code analysis at, 151–155
  - load testing execution at, 312–313
  - profiling tools at, 195–198
  - Team Foundation Server tools at, 397–398
  - Team Foundation version control at, 444–445
- comma-separated value (CSV) files, 293
- comments, 290
- Common Compiler Infrastructure, 157
- Common Object Request Broker Architecture (CORBA), 9–10

- community-driven process templates, 385
  - Comparator, 344
  - comparison reports, 186
  - compile-time reuse, 7–8
  - Completed Work column, 611–613
  - component diagrams
    - creating, 38–43
    - element properties of, 37–38
    - internal component parts in, 43–46
    - introduction to, 33–34
    - in software architecture, 13
    - toolboxes in, 36
    - understanding, 34–36
  - components, 8–9, 34–35
  - concurrency
    - in activity diagrams, 23
    - in check-outs, 390
    - of editing, 388
    - in profiling sessions, 185
    - in profiling tools, 172
  - Configuration Database, 410–411
  - configuration server instances, 410
  - Configure User Tools, 464
  - conflict-resolution mechanisms, 443–444
  - conflicts
    - in branching plan implementation, 461–465
    - channel option for, 434
    - in merges, 462–465
    - in Pending Changes windows, 444
  - Connection Properties dialogs, 210
  - constant loads, 299
  - context parameters
    - defined, 285
    - in manual Web performance tests, 281
    - in Web performance tests, 290–291
  - continuous integration (CI) triggers, 391, 484
  - continuous model of implementation, 564
  - contributor role in team projects, 426
  - Control Map, 343–344
  - controllers
    - configuring, 314–315
    - defined, 313
    - installing, 314
  - copy-and-paste reuse, 8
  - CORBA (Common Object Request Broker Architecture), 9–10
  - counter sets, 303–304
  - CPM (critical path method), 566
  - CPUs (central processing units)
    - Counters property page in, 177–178
    - Sampling in, 171
    - usage of, 188
  - creating
    - class diagrams, 55–57
    - component diagrams, 38–43
    - database development projects, 207–212
    - dependency graphs, 73–74
    - layer diagrams, 82–83
    - load tests, 297–298
    - messages, 29
    - Microsoft Excel Reports, 579
    - Microsoft Office Project projects, 540–544
    - reports using data in OLAP cubes, 577–579
    - reports using work item queries, 579–581
    - Team Foundation Server reports, 575–576
    - team projects, 454
    - tests with virtual environments, 366
    - TFS reports, 588
    - unit tests, 103–106
    - Web performance tests, 281–282
    - ZIP archives, 504–509
  - “Creating and Customizing TFS Reports”, 588
  - critical path method (CPM), 566
  - CruiseControl, 484
  - CSV (comma-separated value) files, 293
  - Current Reports, 580–581
  - customizing
    - build processes, 476, 500–502
    - process templates. *See* process template customizations
    - properties in Unit Test Framework, 119–120
    - reports in Microsoft Excel, 579
  - Cyclomatic Complexity, 163
- ## D
- dashboards
    - defined, 523
    - introduction to, 571
    - publishing Excel reports in, 583–585
    - reports on, 561, 567
    - in Team Foundation Server reporting, 592, 593–595
  - Data Access Layer, 84–85, 87–89
  - Data and Diagnostics tabs, 274, 368
  - Data collection control panels, 175
  - Data Compare tool, 240–241
  - “Data connections have been disabled” warning, 579
  - data flows, 23–25
  - data generation
    - data generators for, 226–227
    - in database development, 224–227
    - introduction to, 224
    - plan for, 224–226
  - data sources, 293–294
  - data tier
    - defined, 380
    - in hosted environments, 419
    - in multi-server installations, 412–413
    - in Team Foundation Server, 407–408
    - in three-tier architecture, 405
  - data tips, 253–254
  - data warehouses, 572–574
  - database administrator (DBA), 206
  - database development. *See also* database testing
    - change management in, 202–203
    - changing schemas in, generally, 215
    - creating projects in, 207–212
    - Data Compare tool in, 240–241
    - data generation in, 224–227
    - deploying changes in, 221–224
    - examining projects in, 212–215

- database development (*continued*)
  - introduction to, 201–202
  - lifecycles, 203
  - offline schema development in, 203–207
  - preservation of intent in, 219–220
  - redistributable deployment engine in, 236–238
  - refactoring databases in, 217–220
  - rename refactoring in, 217–218
  - Schema Compare tool in, 238–240
  - Schema Dependency Viewer, 213–214
  - schema syntax errors in, 216–217
  - Solutions Explorer vs. Schema View, 213
  - testing in. *See* database testing
  - T-SQL file structures in, 214–215
  - T-SQL files, editing directly, 216
  - T-SQL script templates in, 220–221
- Database Scheme Provider, 97
- database testing. *See also* database development
  - advanced unit tests in, 230–231
  - cleanliness in, 232
  - Data Compare tool in, 240–241
  - effectiveness of, 231–233
  - functions in, 227–230
  - introduction to, 201–202, 227
  - redistributable deployment engine in, 236–238
  - related tests, grouping in series, 232–233
  - Schema Compare tool in, 238–240
  - stored procedures in, 227–230
  - triggers in, 227–230
  - T-SQL static analysis in, 233–236
  - unit tests in, 227–233, 261
  - variety of approaches in, 233
- databases
  - binding to, 294
  - in data-driven Web performance tests, 293
  - deployment of, 201–202
  - developing. *See* database development
  - extensibility of, 97
  - servers for, 409
  - testing. *See* database testing
- data-driven tests
  - coded user interface, 347–349
  - in Unit Test Framework, 121–122
  - Web performance, 293–294
- DBA (database administrator), 206
- DCOM (Distributed Common Object Model), 9–10
- debugging. *See also* bugs
  - IntelliTrace for. *See* IntelliTrace
  - My Bugs activity, 332
  - in profiling tools, 198
  - in software development, 98
  - unit tests in Visual Studio Express Edition, 109–110
- decision nodes, 22
- DefaultTemplate process, 493–494
- deleting
  - artifacts in layer diagrams, 86
  - builds, 477
  - data from databases, 226
  - process templates, 631–632
  - shelvetests, 439
  - team projects, 397
- Deming, Edwards W., 562
- dependency graphs
  - creating in Architecture Explorer, 71–73
  - creating without Architecture Explorer, 73–74
  - defined, 15
  - introduction to, 71
  - in layer diagrams, 84
  - legends in, 77–78
  - navigating through, 74–77
  - toolbars in, 78–80
- Dependency Matrix View, 79
- Deploy environment dialogs, 363–364
- deploying database changes, 221–224
- Deployment checkbox, 229
- deployment phase in schema development, 206–207
- deployment scenarios. *See* Team Foundation Server 2010, deploying
- deploy-time reuse, 8–9
- Depth of Inheritance, 163
- description of branches, 457
- “Design Guidelines for Class Library Developers”, 140
- designing visually, 3
- Detail view of load testing results, 312
- Details tab for user story work items, 556
- DEV
  - in advanced branching plans, 468
  - creating, 457–458
  - defined, 453
  - in implementing branching plans, 454–455
  - resolving merge conflicts in, 461–465
- development
  - ASP.NET Development Server, 280–282
  - of databases. *See* database development
  - feature-driven, 568
  - green field, 96
  - Microsoft Developer Division, 383–384
  - model-driven, 3–6, 96
  - of schemas offline, 203–207
  - of software. *See* software development
  - SQL Server Development Center, 587
  - test-driven, 122
- DGML (Directed Graph Markup Language), 15
- DGQL (Directed Graph Query Language), 62–63, 70–71
- diagnostic data adapters
  - introduction to, 262–264
  - settings for, 274
  - in test impact analysis, 275
  - in Test Manager, 322–323
- diagnostic events, 246
- diff-merge tools, 443, 464
- dimensions, defined, 575
- direct links queries, 536
- Directed Graph Markup Language (DGML), 15
- Directed Graph Query Language (DGQL), 62–63, 70–71
- directed graphs. *See* dependency graphs
- disabling rules, 149
- Distributed Common Object Model (DCOM), 9–10

distributed components, 9–10

distributed load testing

- agents, configuring, 315
- agents, installing, 314
- attributes in, 315
- controllers, configuring, 314–315
- controllers, installing, 314
- defined, 278
- introduction to, 313–314
- IP switching in, 315
- running, 316
- settings for, 315–316
- viewing, 317
- weighting agents, 315

distributed services, 10–11

DLLs (dynamic-link libraries), 8

document libraries, 383

Documents link, 595

Documents node, 527

domain-specific languages (DSLs)

- introduction to, 3
- in Microsoft’s modeling strategy, 4
- for software architecture, 6–7

drop folder locations, 487

drop locations, 478

DSLs (domain-specific languages). *See* domain-specific languages (DSLs)

Dynamic Ports property, 281

dynamic-link libraries (DLLs), 8

**E**

earned value (EV), 565

editing

- concurrency of, 388
- load testing, 306–312
- performance tests, 289
- process templates. *See* Process Template Editor
- test lists. *See* Test List Editor
- Web tests. *See* Web Test Editor
- XML, 620–621

effective database testing, 231–233

Eisenhower, Dwight D., 529

EJBs (Enterprise Java Beans), 9

elapsed time, 187

element properties, 37–38

e-mail alerts, 492

enabling

code analysis tools, 144–146

code coverage, 129–130

content in reports, 579

end-to-end quality, 552

enhanced assertion reporting, 350–351

EnterDataAndClickAdd, 346, 350

Enterprise Java Beans (EJBs), 9

enterprises, 417–418

enumerations, 47–48

Environment Viewer, 364–365, 369–370

epics, 600

Equals operator, 114

Error, 145–146

Errors tables, 311–312

estimations, 608, 611–612

EV (earned value), 565

event logs, 263

events, 246–248

Excel. *See* Microsoft Office Excel

Excel Reports. *See* Microsoft Office Excel Reports

exceptions, 247

exclusive time, 188

“Execute unit tests using the following data connection”, 228

execution occurrences, 29

Expand Wildcards refactoring, 218–219

Expected Results, 328

ExpectedException attribute, 119

exporting

- breakpoints, 251–252
- data tips, 254
- test results, 271

extensibility hooks, 400

external assemblies, 75–76

external interfaces, 45

extraction rules, 290

eXtreme Programming (XP), 484, 551

**F**

F5 keys, 472

failure

- of assertions, 116
- of check-in policies, 436

of requests, 288–289

of unit tests, 105–106

fast-forward for manual testing, 333

FDD (Feature-Driven Development), 551, 568

FI (forward integration). *See* forward integration (FI)

field mapping types, 539

field types, 539

fields, 5

file labels, 437–438

filtering options, 63–64

Fixed run count, 286

FIXES, 468–469

flat list queries, 536

focus on customers, 552

folder mappings, 426–427, 486

follow history, 390

font properties, 56

fork nodes, 23

Form Post Parameters, 294

forward integration (FI)

- in advanced branching plans, 469
- defined, 449
- in Source Control Explorer, 440

found messages, 29

Fowler, Martin, 484, 600

FQDNs (fully qualified domain names), 374–375

frequency of deployment, 552

full test settings, 323

fully qualified domain names (FQDNs), 374–375

Fully Qualify Name refactoring, 218–219

functions, defined, 227–230

Functions class, 126–129

Functions Details view, 191–192

“Functions Doing Most Individual Work”, 189

Functions view, 191–192

FunctionsTest.cs, 103–104

FxCop

- as code analysis tool, 140–141
- options in, 151–154
- project files in, 154–155

## G

Gantt views, 540  
 garbage collection, 194  
 gated check-ins  
   in builds, generally, 475  
   trigger controls for, 391  
   triggers for, 485  
 General section  
   of Intellitrace, 245  
   in Performance Explorer session  
     properties, 174–175  
   in Team Foundation Build, 482–483  
 generalist testers, 259, 319  
 Generate Code, 343–346, 353–354  
 Generate DROP statements, 223–224  
 “Generate test data before unit tests  
 are run”, 229  
 generations, 194  
 generic tests, 261  
 Get Version, 500  
 Get Work Items, 540–541, 545  
 GetRequestEnumerator method,  
   295–296  
 GlobalSuppressions.cs, 151  
 goal-based loads, 299, 307–308  
 golden images, 359  
 GotDotNet.com, 140  
 graphs view, 309–310  
 green field development, 96  
 groups  
   in Architecture Explorer, 66–68  
   in layer diagrams, 83  
   namespaces in, 66  
   in Process Template Editor, 627  
   in team projects, 386  
 Guckenheimer, Sam, 549  
 guidance, 632

## H

Hallberg, Aaron, 495  
 hardware requirements, 412–413  
 helper code, 150  
 hidden fields, 291  
 History windows, 436–437  
 hooks, 35  
 hosting

environments, 418–419  
 groups, 358  
   in Team Foundation Server, 402  
 Hot Path, 189, 193  
 hotspots, 168, 184  
 HttpContext.Items, 290  
 hubs, 66, 77

## I

IDEs (integrated development  
 environments). *See* integrated  
 development environments (IDEs)  
 IIntrospectionRule interface, 159  
 impact analysis. *See* test impact  
 analysis  
 Implementation tab, 557  
 implementations, defined, 10  
 Import Database Schema page,  
   209–210  
 importing  
   breakpoints, 252  
   data tips, 254  
   test results, 270  
 inbound links, 65–66  
 Inbound Navigation, 66–69  
 inclusive time, 188  
 Indent option, 610  
 independent software vendors  
   (ISVs), 384  
 individual capacity planning,  
   612–613  
 individuals, 416  
 inheritance  
   in Architecture Explorer, 56  
   in code analysis, 145  
   depth of, 163  
 initial nodes, 22  
 initialization, 110–113  
 in-place upgrades, 420  
 Input list, 545–547  
 input pins, 24  
 installing. *See also* Team Foundation  
   Server Installation Guide  
   agents, 314  
   Build services, 478  
   controllers, 314  
   Process Template Editor, 622–623  
   instances, 194–195, 409  
   instrumentation  
     code coverage and, 129, 199  
     in profiling tools, 168–169, 171  
     properties of, 181  
     sessions, configuring, 184  
   instrumented profiling sessions, 194  
   instrumenting applications, 168  
 integrated development  
   environments (IDEs)  
     code analysis in, 140–141  
     introduction to, 4  
     profiling tools integrated with. *See*  
       profiling tools  
 integrating  
   build process and code analysis  
     tools, 155  
   changes. *See* merging  
   software components, 472  
   testing in software lifecycles, 259  
 Intellitrace  
   Advanced option in, 245–246  
   breakpoints in, 251–252  
   calls history in, 250  
   debugging with, generally, 243–244  
   events in, 246–248  
   General section of, 245  
   introduction to, 243  
   modules in, 246–247  
   navigation bar in, 249–250  
   Options in, 244–245  
   pinnable data tips in, 253–254  
   playback in, 248–249  
   in software development, 98  
   in software testing, 262–263  
   trace log files in, 250  
 interaction  
   property pages, 176–177  
   uses, 29  
   worksheets, 603  
 interfaces  
   in class diagrams, 46–47, 51–53  
   in component diagrams, 34, 40–46  
   defined, 10  
 internal component parts, 43–46  
 Internet Explorer, 283  
 Interruptions worksheets, 603–609  
 introspection, 155–156

- IP switching, 315
  - issues
    - in profiling tools, 198–199
    - spreadsheets, 613
    - work items, 559
  - ISVs (independent software vendors), 384
  - iteration backlog workbooks
    - capacity planning worksheets in, 612
    - introduction to, 608
    - iteration backlog worksheets in, 609–612
    - locating, 608–609
    - team capacity worksheets in, 612
    - team member capacity worksheets in, 613
  - iterations
    - backlog workbooks. *See* iteration backlog workbooks
    - planning, 561, 607–608
    - in team projects, 383
    - worksheets, 606–607
  - Iterations tab, 528
  - iterative development, 204–205
- J**
- JAR (Java Archive) files, 8
  - Java, 399
  - Java Archive (JAR) files, 8
  - JavaScript, 196–197
  - join nodes, 22
  - “Just My Code”, 198
- K**
- “Keep Target Branch Version”, 465
- L**
- lab agents, 359
  - Lab center, 321
  - Lab Management
    - agents in, 359
    - automated build-deploy-test in, 371–375
    - creating new test settings in, 366–369
    - golden images in, 359
    - infrastructure of, 358–359
    - introduction to, 357–358
    - Lab Workflow Parameters wizard in, 372–375
    - manual tests in, 369–371
    - physical environments in, 375
    - virtual environments, overview of, 360–366
    - virtual environments, testing with, 366–371
  - Lab plug-ins, 618
  - lab processes, 386
  - labeling
    - breakpoints, 252
    - files, 437–438
    - in Team Foundation Server, 390
  - Lamb, Jim, 513
  - late binding, 155
  - Launch property page, 175–176
  - layer diagrams
    - adding multiple objects to, 84–85
    - build process and, 90–91
    - creating, 82–83
    - defining dependencies in, 86–88
    - defining layers on, 83–84
    - introduction to, 81–82
    - Layer Explorer and, 85–86
    - layers for single artifacts in, 84
    - in software architecture, 13
    - validating, 88–90
  - Layer Explorer, 85–86
  - layers, defined, 81
  - LCLs (Lower Control Limits), 562–563
  - lead role in team projects, 426
  - left navigation panes
    - dashboards in, 593–595
    - Documents link in, 595
    - Excel Reports link in, 595
    - introduction to, 592
    - process guidance link in, 595
    - Reports link in, 595
    - in Team Foundation Server reporting, 592–595
    - Team Web Access link in, 593
  - Level 3 compliance, 564–566
  - Level property, 292–293
  - libraries
    - document, 582–583
    - dynamic-link, 8
    - of tests, 97
    - wiki, 560, 632
    - of workflow activities, 513
  - Library projects, 141–149
  - library servers, 358
  - library shares, 358
  - licensing, 392
  - lifelines, 28–31
  - lightweight test settings, 323
  - Lines of Code, 164
  - Link Category options, 78
  - Link Property options, 78
  - link types, 531–533, 626–627
  - linked artifacts, 85–86
  - links, 61, 65–66
  - Links and Attachments, 541
  - literals, 48
  - Load Test Editor, 306–312
  - load testing. *See also* Web performance testing
    - browser mixes in, 303
    - command-line execution of, 312–313
    - creating and configuring, 297–298
    - in database testing, 233
    - Detail view of results, 312
    - distributed. *See* distributed load testing
    - editing, 306–312
    - Errors table in, 311–312
    - executing, 308
    - goal-based load profiles in, 307–308
    - graphs view of results of, 309–310
    - introduction to, 297
    - load patterns in, 299–300
    - network mixes in, 302–303
    - Pages tables in, 310
    - performance counter sets in, 303–304
    - results of, 308–312
    - run settings for, 304–306
    - scenarios in, 298, 306
    - in software testing, 261
    - SQL Tracing in, 307, 311
    - storing load test run data, 308
    - tables view of results of, 310
    - test mix model, 300–302

load testing (*continued*)  
 test tables in, 310  
 think times in, 298–299  
 Thresholds tables in, 311  
 transactions tables in, 310–311

Local test settings, 274

locals windows, 253

lock options, 435

logging

in builds, 481

in Intellitrace, 245

in Team Foundation Build, 495

in Unit Test Framework, 112

logical class diagrams. *See* class diagrams

lollipops, 35

LOP (Low-Overhead Provider), 169

lost messages, 29, 31

Lower Control Limits (LCLs),  
 562–563

Low-Overhead Provider (LOP), 169

## M

Machine properties, 361–362

### MAIN

in advanced branching plans, 468  
 converting folders to branches in,  
 455–456

creating DEV branches from, 457–458  
 defined, 453

in implementing branching plans,  
 454–455

resolving merge conflicts in,  
 461–465

Maintainability Index, 164

Manage Build Controllers, 517

managed code debugging, 246

managed code profiling, 169

Managed Extensibility Framework  
 (MEF), 98

managing

labs. *See* Lab Management

process templates. *See* Process  
 Template Manager

projects. *See* project management  
 releases, 564

session reports, 186–187

software configuration. *See*

Software Configuration  
 Management (SCM)

tests. *See* Test Manager

unit tests, 106–108

virtual machines, 358–362

manual build trigger controls, 391

manual software testing, 259, 261

manual testing

assigning configurations in, 330

assigning testers in, 330

authoring test cases in, 327–329

automated, 336–338

builds in, 323–324

configurations of tests in, 325–326

of databases, 233

impact analysis in, 324–325

introduction to, 319

plan contents in, 326–330

results tracking, 330–332

running, 330–332

saving results, 335–336

settings for, configuring, 322–323

shared steps in, 329–330

supported technologies in, 335

Test Manager for, 319–320

test plans in, generally, 320–322

Test Runner for, 332–335

of Web performance, 281

Manual triggers, 483

mapping

fields, 539

folders, 486

to local folders, 426–427

maturity capability levels. *See*

MSF for CMMI Process  
 Improvement v5.0

MDA (Model-Driven Architecture), 5

MDD (model-driven development).

*See* model-driven development  
 (MDD)

measures, defined, 575

MEF (Managed Extensibility

Framework), 98

members, 68–69

memory profiling reports, 190

merge nodes, 22

Merge Tool, 462–465

Merge Wizard, 443, 458–462

merging. *See also* branching  
 for branching plan implementation,  
 458–462

defined, 449

resolving conflicts in, 461–465

in Software Configuration  
 Management, 447–448

in Source Control Explorer,  
 443–444

in Team Foundation Server, 389

in Team Foundation version  
 control, 423–424, 440

understanding, 448–449

methods, 5, 72–75

Microsoft Developer Division,  
 383–384

Microsoft Office

Excel in. *See* Microsoft Office Excel

Excel Reports in. *See* Microsoft  
 Office Excel Reports

introduction to, 400

Project in. *See* Microsoft Office

Project

for project management, 538–544

SharePoint in, 401, 528

Team Foundation Server and,  
 538–544

Microsoft Office Excel

Input list option in, 545–547

iteration backlog workbooks in, 608

planning projects with, 545

product planning workbooks in,  
 602–605

for project management, 544–547

Team Foundation Server and,  
 395–396, 544–547

Work Item Add-in, 538

Microsoft Office Excel Reports

creating, 579–581

customizing reports in, 579

introduction to, 575–576

OLAP cubes in, 577–579

publishing, generally, 581–582

publishing to document libraries,  
 582–583

publishing to Excel Services,  
 583–585

- in Team Foundation Server
    - reporting, 595
    - work item queries in, 579–581
    - working with, generally, 576
  - Microsoft Office Project
    - creating projects in, 540–544
    - field types in, 539
    - introduction to, 539
    - for project management, 539–544
    - Team Foundation Server vs., 397, 542–544
  - Microsoft Office SharePoint Server (MOSS). *See also* SharePoint
    - in Team Foundation architecture, 414
    - Team Foundation Server 2010 and, 401–402
    - team projects in, 528
    - Windows SharePoint Services
      - vs.. *See* Windows SharePoint Services (WSS)
  - Microsoft Outlook, 529
  - Microsoft partners and community, 568–569
  - Microsoft Patterns & Practices: Agile Development Showcase*, 552
  - Microsoft Solutions Framework, for Agile Software Development. *See* MSF for Agile Software Development v5.0
  - Microsoft Solutions Framework, for Capability Maturity Model Integration. *See* MSF for CMMI Process Improvement v5.0
  - Microsoft SQL Server, 202, 424
  - Microsoft Symbol Server, 198
  - Microsoft System Center Virtual Machine Manager (SCVMM), 358–362
  - Microsoft Team Explorer, 490
  - Microsoft Team Manager, 132
  - Microsoft Test Manager
    - action logs and recordings in, 369
    - activity centers in, 321
    - automated tests in, 337–338
    - configuring manual tests in, 366–367
    - diagnostic data adapters in, 322–323
    - executing load tests in, 308
    - introduction to, 319–320
    - for software testing, generally, 264
    - templates in, 361
    - test plans in. *See* test plans
  - Microsoft Test Runner
    - action recordings in, 333–335
    - bugs in, 336, 369–371
    - for manual tests, 332–335
  - Microsoft Visual SourceSafe (VSS), 388–389, 424–425
  - Microsoft Visual Studio 2010
    - Ultimate. *See* Visual Studio 2010 Ultimate
  - `Microsoft.Cci` assembly, 157–158
  - migration, 421, 553–554
  - minimally credible releases, 601
  - Model-Driven Architecture (MDA), 5
  - model-driven development (MDD), 3–6
  - modeling projects, 82–83, 90
  - modules, 246–247
  - MOSS (Microsoft Office Sharepoint Server), 401, 581–583
  - Move to Schema refactoring, 218
  - MSBuild, 474–475
  - MSDN documentation, 196
  - MSDN forums, 415
  - MSF for Agile Software Development v5.0
    - agile planning workbooks in, 561
    - bug work items in, 557–559
    - folder view of process template in, 550
    - introduction to, 551–553
    - issue work items in, 559
    - migrating to process templates in, 553–554
    - new features of, 401, 553
    - planning workbooks in. *See* agile planning workbooks
    - process guidance in, 560
    - process templates in, 385, 525, 551
    - reports in, 560–561, 589, 629
    - samples in, 562
    - shared steps work items in, 560
    - task work items in, 557
    - templates in, 562
    - test case work items in, 559
    - user story work items in, 555–557
    - work items in, 529–530, 554–560, 621
  - MSF for CMMI Process Improvement v5.0, 525
    - architects and, 565
    - business analysts and, 564
    - CMMI Level 3 vs., 564–568
    - developers and, 565
    - introduction to, 562–565
    - link types in, 626–627
    - process guidance in, 567
    - process templates in, 385, 551, 565–568
    - projects in, 564
    - release managers and, 564
    - reports in, 567–568, 589
    - testers and, 565
    - work items in, 566
  - `MSTest.exe`, 273, 312–313
  - multi-server installations, 412–413, 417–418
  - My Dashboard, 594–595
  - `My Queries`, 535
- ## N
- namespaces, 63–66
  - naming conventions, 154
  - navigation
    - in Architecture Explorer, 62–64
    - gutters, 246
    - in Intellitrace, 249–250
  - Neighborhood Browse Mode, 79–80
  - .NET Memory Allocation
    - profiling collection, 175
    - in profiling tools, generally, 172
    - session configuration, 184–185
  - network emulation data adapters, 263
  - network mixes, 302–303
  - network speeds, 287
  - New Database Project Wizard, 208–212
  - New Load Test Wizard, 297–305
  - New Team Project Wizard, 525–527
  - New Test Data Source Wizard, 347–348

New Work Item Report window,  
580–581

Newkirk, Jim, 122

no branching, 449–450

No Repro bug, 262

Node Category options, 77

Node Navigation

classes in, 66–67

members in, 68–69

namespaces in, 65–66

Node Property options, 77

nodes, 61–62

Notepad, 293, 620

novels, 600

NUnit, 102

## O

Object Management Group (OMG), 5

object nodes, 23

object-orienting programming  
(OOP), 7

objects, 7–8

Objects Lifetime view, 194–195

observing test execution and results,  
287–289

offline, defined, 204

offline schema development

build output in, 206–207

deployment phase in, 206–207

first steps in, 204

introduction to, 203

iterative development in, 204–205

schema testing in, 205–206

OLAP cubes, 572–575

OMG (Object Management Group), 5

OOP (object-orienting  
programming), 7

open communications, 552

Open Source

7-zip archives in, 504–507

build servers in, 484

process templates in, 385

operational stores, 572–573

operations

in class diagrams, 47–48, 52–53

defined, 5

options

in Architecture Explorer,  
generally, 62

for classes, 66–68

for debugging, 244–245

for members, 68–69

for namespaces, 64–66

ordered software tests

creating, 272

defined, 261, 265

executing and analyzing, 273–274

introduction to, 272

properties of, 273

Ordered Test for databases, 232

outbound links, 65–66

Outbound Navigation

classes in, 67–68

members in, 69

namespaces in, 66

Outdent option, 610

output pins, 24

Override Project Settings, 180

overriding check-in policies, 436

owners, 456, 483

## P

Package Binaries, 504–508, 510

Pages tables, 310

parallel development, 204

Parameter Values, 329

parameterizing Web servers, 285

parent folders, 443

parent-child relationships

Add Child option in Excel, 610

in project management, 532–533

in user stories, 556

of work items, in Excel, 546–547

of work items, in Team Explorer,  
541–542

parts, 44

patches, 467

path to publish symbols, 497

paths, 421

pausing Test Builder, 342

Pending Changes window

in branching plan implementation,  
460–463

channels in, 433–434

Conflicts channel option in, 444

merge conflicts in, 462–465

merged changes in, 460–461

shelvesets in, 439

user tools in, 464

“People come first”, 552

per feature branching, 451–452

per release branching, 450–451

Perez, Juan, 549

performance analysis. *See* profiling

tools; Web performance testing

performance counter sets, 303–304

performance counters, 310

Performance Explorer

Allocation view in, 193–194

Call Tree view in, 193

Caller/Callee view in, 192–193

concurrency profiling sessions in, 185

CPU Counters property page in,

177–178

Functions Details view in, 191–192

Functions view in, 191–192

general property pages in, 174–175

general session properties in, 174

instrumentation sessions in, 184

Interaction property page in,

176–177

introduction to, 167–168, 173

Launch property page in, 175–176

.NET Memory Allocation sessions  
in, 184–185

Objects Lifetime view in, 194–195

performance sessions in, 171–173,

185–186

report information, generally,

187–188

sampling sessions in, 182–184

session reports, managing, 186–187

session reports, reading and

interpreting, 187

session targets, configuring,

179–182

Summary view in, 188–191

views, generally, 187–188

Windows Counter property page

in, 178–179

Windows Events property page

in, 178

- Performance Wizard, 171–172
  - permissions
    - in branching plan implementation, 466–467
    - in database development, 212
    - for PROD access, 466–467
    - in product planning workbooks, 603
    - in team projects, 386
    - in workspaces, 431
  - Personal Web Site Starter Kit, 279
  - perspectives, 578–579
  - physical environments, 375
  - pinnable data tips, 253–254
  - planning
    - branching. *See* branching plan implementation
    - data generation, 224–226
    - manual testing, 326–330
    - project management, 528–529
    - projects, 545
    - scenarios, 453–454, 468
    - tests, 326–330
    - workbooks, 523
  - Planning Extreme Programming*, 600
  - planning poker, 608
  - platform-specific models (PSMs), 5
  - playback, 248–249
  - plug-ins, 618–620
  - Policy Warnings channel option, 434
  - Polytron Version Control System (PVCS), 388
  - port numbers, 282
  - Portal option, 628–629
  - Portal plug-ins, 618
  - portals
    - introduction to, 571
    - of projects. *See* project portals as SharePoint sites, 583
    - team project, 528
  - power tools, 622
  - predecessor-successor links, 533
  - pre-defined work item queries, 535–537
  - prerequisites, 132
  - preservation of intent, 219–220
  - Pressman, Roger, 448
  - private builds, 475, 491
  - private members, 122
  - private workspaces, 432
  - `PrivateObject`, 123–125
  - `PrivateType`, 125–126
  - probes, 168
  - process areas, 564
  - process arguments, 509
  - process guidance
    - in Agile Software Development, 560
    - in CMMI Level 3, 567
    - defined, 523
    - links, 595
    - in process templates, 567
  - Process Improvement. *See* MSF
  - for CMMI Process Improvement v5.0
  - process parameters, 509–512
  - process template customizations. *See also* process templates
    - contents in, 617–618
    - deleting, 631–632
    - downloading to desktops, 616–617
    - guidance for, 632
    - introduction to, 615–616
    - online information about, 632
    - plug-ins, 618–620
  - Process Template Editor for. *See* Process Template Editor
  - tools for, generally, 620
  - uploading process templates, 631
  - wiki libraries on, 632
  - `witadmin` command-line utilities for, 621
  - XML Editor for, 620–621
- Process Template Definition, 617
  - Process Template Editor. *See also* Process Template Manager
    - Areas and Iterations section in, 627
    - customizations in. *See* process template customizations
    - Groups and Permissions section in, 627
    - installing, 622–623
    - introduction to, 622
    - link types in, 626–627
    - Portal option in, 628–629
    - Reports node in, 629
    - source control settings in, 627
    - work item tracking in, 624–626
  - work items on existing team
    - projects, editing, 629–631
  - working with, generally, 623
- Process Template Manager. *See also* Process Template Editor
    - introduction to, 616
    - New Project Wizard in, 618
    - upload/download operations in, 616–617
  - process templates
    - architects and, 565
    - for builds, 492–500
    - business analysts and, 564
    - CMMI Process Improvement vs., 564–568
    - customizing. *See* process template customizations
    - defined, 523
    - developers and, 565
    - editing. *See* Process Template Editor
    - introduction to, 549–550
    - managing. *See* Process Template Manager
    - from Microsoft partners and community, 568–569
    - overview of, 550–551
    - plug-ins, 618–620
    - process guidance in, 567
    - projects in, 564
    - release managers and, 564
    - reports in, 567–568
    - in Team Foundation Server, 385–386, 551
    - in team projects, 523–525
    - testers and, 565
    - upgrading, 421
    - work items in, 566
  - `ProcessClasses` method, 192–193
  - processes, defined, 487–488, 549
  - `processtemplate.XML` (Extensible Markup Language), 619
  - PROD
    - access to, 466–467
    - in advanced branching plans, 469
    - code stabilizing into, 465–466
    - defined, 454

- product backlog worksheets,
    - 603–606
  - product backlogs, 600–601
  - product planning workbooks
    - agile planning workbooks, 602–607
    - interruptions worksheets in, 607
    - introduction to, 602
    - iterations worksheets in, 606–607
    - locating, 602
    - product backlog worksheets in, 603–606
    - setting up, 602–603
  - profiler enhancements, 97
  - profiling tools
    - blank performance sessions, adding, 172–173
    - command-line utilities, 195–198
    - debugging symbols in, 198
    - instrumentation and code coverage in, 199
    - introduction to, 167–168
    - issues in, 198–199
    - for JavaScript, 196–197
    - Just My Code, 198
    - Performance Explorer. *See* Performance Explorer
    - Performance Explorer performance sessions, creating, 171–173
    - Performance Wizard, 171–172
    - sample applications, creating, 169–171
    - types of, 168–169
    - using, generally, 169
    - for virtual machines, 196
    - in Visual Studio 2010, 169
  - Project Dashboard, 594–595. *See also* dashboards
  - project files, 154–155
  - project management
    - creating projects, 540–544
    - field types in, 539
    - Input list option in, 545–547
    - introduction to, 521–522
    - link types in, 531–533
    - Microsoft Office Excel for, 544–547
    - Microsoft Office for, 538–544
    - Microsoft Office Project for, 539–544
    - new team projects in, 525–527
    - overview of, 522–523
    - parent-child links in, 532–533
    - planning in, 528–529, 545
    - predecessor-successor links in, 533
    - pre-defined work item queries in, 535–537
    - related items links in, 532
    - reports in, 567
    - structuring projects in, 527–528
    - Team Foundation Server in, 527–528
    - team project portals in, 528
    - team projects, creating, 523–527
    - work item queries in, 535–538
    - work items and link types in, 531–533
    - work items, creating and updating, 534–535
    - work items in, generally, 529–531
  - Project planning workbooks, 561
  - project portals. *See also* portals
    - dashboards in, 593–595
    - Documents link in, 595
    - Excel Reports link in, 595
    - left navigation panes in, 592–595
    - process guidance link in, 595
    - Reports link in, 595
    - in Team Foundation Server reporting, 592
    - Team Web Access link in, 593
    - top horizontal menus in, 595–596
  - projects
    - in CMMI Level 3, 564
    - Library, 141–149
    - managing. *See* project management
    - Microsoft Office. *See* Microsoft Office Project
    - modeling, 82–83, 90
    - planning workbooks, 561
    - portals. *See* project portals
    - in process templates, 564
    - publishing, 542
    - Report Server Project, 587
    - team. *See* team projects
    - test. *See* test projects
  - promotion level branching, 451
  - properties pages
    - CPU Counters, 177–178
    - General page in, 174
    - Launch, 175–176
    - properties exposed for common customizations, 476
    - Tier Interaction, 176–177
    - Windows Counters, 178–179
  - Properties window, 289
  - protocol support, 297
  - provided interfaces, 34, 45
  - proxies, 390
  - PSMs (platform-specific models), 5
  - public workspace, 431
  - Publish and Refresh, 604–605
  - publishing, 542, 581–585
  - PVCS (Polytron Version Control System), 388
- Q**
- quality gates, 552
  - queries, 69–71
  - Query List, 545
  - query-based test suites, 327
  - queued builds, 479–480, 489–491
  - quick lists, 253
- R**
- Rational Unified Process, 4
  - RDL (Report Definition Language) reports. *See* Report Definition Language (RDL) reports
  - read-only permissions, 466–467
  - Reason field, 558
  - recorded Web performance testing, 282–284, 289
  - RedGate, 163
  - redistributable deployment engines, 236–238
  - redundant test settings, 313
  - refactoring
    - logs, 219–220
    - static analysis rules, 235–236
    - in unit testing, 129
  - references, 114
  - reflection
    - in code analysis tools, 155–156
    - in unit testing, 103, 123–125

- Reflector, 163
  - refresh intervals, 573–574
  - regression testing, 101, 327
  - related items links, 532
  - related test series, 232–233
  - Release Configuration mode, 185
  - releases
    - defined, 449
    - managers of, 564
    - planning, 600–601, 607
  - Remaining Work column, 611–613
  - remote machines, 275, 304
  - rename refactoring, 217–218
  - renaming, 219
  - Report Builder, 586–587
  - Report Definition Language (RDL)
    - reports. *See also* reports
    - introduction to, 585–586
    - Report Builder in, 586–587
    - Report Designer in, 587–588
    - in Team Foundation Server
      - reporting, 585–588
  - Report Designer, 575–576, 587–588
  - Report Server Project, 587
  - Reports, 173
  - reports
    - in Agile Software Development, 560–561
    - in CMMI Level 3, 567–568
    - defined, 523
    - in Performance Explorer, 187–188
    - plug-ins for, 618
    - in process templates, 567–568
    - in Report Definition Language. *See* Report Definition Language (RDL) reports
    - in Team Foundation Server. *See* Team Foundation Server reporting
    - in team projects, 383, 386
  - Reports folders, 186
  - Reports link, 595
  - Reports node, 527, 629
  - repository folders, 426
  - request properties, 289
  - required interfaces, 34, 40–42
  - requirements-based test suites, 326–327
  - Resolution, 161–162
  - Resolved work items, 556–558
  - resource contention, 185
  - results
    - expected, 328
    - of load testing, 308–312
    - of manual tests, 330–332, 335–336
    - observing test execution and, 287–289
    - of performance tests, 313
    - publishing, 271
    - tools for. *See* Test Results
  - retention policies, 488–489
  - retrospectives, 613
  - reverse integration (RI)
    - in advanced branching plans, 469
    - defined, 449
    - in Source Control Explorer, 440
  - RI (reverse integration). *See* reverse integration (RI)
  - right-clicking, 127
  - rigs, 314–316
  - risk vs. productivity, 447
  - Risk work items, 624–626
  - Roeder, Lutz, 163
  - roles
    - in Lab Management, 367
    - properties of, 54–55
    - security, 425–426
    - in software testing, 260
    - in Web sites, 279–280
  - rolling builds, 391, 484
  - rule descriptive XML, 157. *See also* XML (Extensible Markup Language)
  - rules in code analysis
    - creating, 155–157
    - deploying, 162–163
    - existing, 163
    - for implementation, 157–161
    - Rules.xml, 161–162
    - violation of, 147–151
  - run settings, 284, 304–306
  - running tests
    - coded user interface, 347
    - directly from code, 108
    - distributed load, 316
    - manually, 330–332
    - manually, with environments, 369–371
    - unit, 108
    - Web performance, 287
  - run-time reuse, 9–10
- S**
- Samples and Templates links, 595
  - sampling
    - in Agile Software Development, 562
    - profilers, 168–169
    - rates, 305
    - sessions, 182–184
  - Scalar Value test conditions, 230
  - Scaling Team Foundation Server 2010, 409
  - scenarios, 298, 306
  - schedules as triggers, 391, 485
  - schema
    - in database testing, 205–206
    - offline development of, 203–207
    - syntax errors in, 216–217
    - testing, 205–206
  - Schema Compare tool, 238–240
  - Schema Dependency Viewer, 213–214
  - SCM (Software Configuration Management), 425
  - scouting builds, 323
  - Scrum, 551, 568
  - Scrum Task Board, 556
  - (SCVMM) System Center Virtual Machine Manager, 358–362
  - SCVs (special cause variations), 562
  - security
    - groups, 386
    - plug-ins, 618
    - roles, 425–426
    - warnings, 579
  - Selection menus, 76
  - self messages, 29
  - sequence diagrams
    - creating, 30–31
    - introduction to, 28
    - in software architecture, 12
    - toolboxes in, 29–30
    - understanding, 28–29
  - sequential test order, 301

- Server, Team Foundation. *See* Team Foundation Server 2010
- server projects, 209
- service-oriented architectures (SOAs), 3
- services, 10–11
- session reports, 186–187
- session targets, 179–182
- Set Database Options page, 209
- settings
  - in automated software testing, 274–275
  - for diagnostic data adapters, 274
  - for distributed load testing, 315–316
  - full, 323
  - Local, 274
  - for manual testing, 322–323
  - Override Project Settings, 180
  - read-only, 627
  - in software testing, 274–275
  - Source Control Settings, 627
  - Trace and Test Impact, 274
  - for Web performance tests, 285–287
  - worksheets for, 609
- Shared Documents links, 595
- shared steps
  - in manual testing, 329–330
  - in test plans, 329–330
  - work items, 560
- shared vision, 552
- SharePoint
  - dashboards in, 593
  - in hosted environments, 418
  - in Microsoft Office. *See* Microsoft Office SharePoint Server (MOSS)
  - new team projects in, 523–526
  - publishing Excel reports in, 581–583
  - storing worksheets in, 607
  - in Team Foundation architecture, 408–411, 414
  - team project portals in, 414
  - team projects in, 386
  - in Windows. *See* Windows SharePoint Services (WSS)
- sharing breakpoints, 251–252
- SharZipLib, 513
- shelvesets, 428, 439
- shelving
  - code, 423–424
  - in Team Foundation Server, 389–390, 438–439
- Show Hubs analyzer, 77
- Show Reflexive View, 79
- signed assemblies, 181
- Simple Object Access Protocol (SOAP), 10
- SimpleWPFCalculator, 340–341
- Simulate think times, 287
- single-server installations, 412, 420
- small shops, 416–417
- small teams, 416
- smart tags, 41
- SMEs (subject matter experts), 559
- snapshots, 365–366, 370–371
- SOAP (Simple Object Access Protocol), 10
- SOAs (service-oriented architectures), 3
- software architecture. *See also* software development
  - activity diagrams in, 11–12
  - Architecture Explorer in, 14–15
  - class diagrams in, 13
  - compile-time reuse in, 7–8
  - component diagrams in, 13
  - components in, 8–9
  - deploy-time reuse in, 8–9
  - designing visually, 3–4
  - distributed components in, 9–10
  - distributed services in, 10–11
  - domain-specific languages for, 6–7
  - introduction to
  - layer diagrams in, 13
  - modeling strategy for, 4–6
  - new tools for, 10–15
  - objects in, 7–8
  - run-time reuse in, 9–10
  - sequence diagrams in, 12
  - services in, 10–11
  - use case diagrams in, 11
- Software Configuration Management (SCM), 425
- software development
  - agile. *See* MSF for Agile Software Development v5.0
  - architecture in. *See* software architecture
  - code analysis in, 97
  - database extensibility in, 97
  - debugging in, 98
  - IntelliTrace in, 98
  - introduction to, 95–98
  - new features for, 96
  - profiler enhancements in, 97
  - test-first experience in, 98
  - testing. *See* software testing
- Software Engineering: A Practitioner’s Approach*, 448
- Software Engineering with Microsoft Visual Studio Team System*, 549
- software factories, 5
- Software Factories: Assembling Applications with Patterns, Works, Models and Tools*, 5
- software requirements, 414
- software testing. *See also* software development
  - automated management of, 265
  - categories in, 267
  - diagnostic data adapters in, 262–264
  - introduction to, 259–260
  - role-based, 260
  - settings in, 274–275
  - test impact analysis in, 97, 275–276
  - Test Manager for, 264
  - test projects in, 265–267
  - Test Results in, 269–271
  - Test View in, 267–269
  - types of, 260–262
- solution configurations, 495
- Solution Explorer
  - activity diagrams in, 26
  - adding projects to source repository in, 432
  - build process and, 90
  - class diagrams in, 55
  - coded user interface tests in, 341
  - component diagrams in, 38, 43

- database changes in, 221–224, 237–238
  - layer diagrams in, 84–85
  - Schema View vs., 213
  - sequence diagrams in, 30
  - software testing in, 274
  - T-SQL static analysis in, 233
  - unit tests in, 108, 132–136, 229–231
  - use case diagrams in, 21
  - Solution to Version Control, 454–455
  - source branches, 459–460, 467
  - source code files, 433
  - source control. *See also* Team Foundation version control for check-in policies, 435–436 explorer. *See* Source Control Explorer
    - in new team projects, 525–527
    - in Process Template Editor, 627
  - Source Control Explorer
    - branching plans in, 454–458, 465–466
    - build process templates in, 501–504
    - checking out files from, 434–435
    - labeling files in, 437–438
    - merging in, 443, 458–462
    - navigating branches in, 440–442
    - PROD access in, 465–467
    - in Team Foundation version control, 427–432
    - View History in, 436–437
  - Source Files channel option, 434
  - special cause variations (SCVs), 562
  - specialist testers, 260
  - specifying branch owners, 456
  - spikes, 608
  - sprints, 383, 607
  - SQL Reporting Services Reports, 575, 589–590
  - SQL Server
    - 2008, 414
    - Configuration Database in, 410–411
    - Development Center, 587
    - in multi-server installations, 412
    - in Team Foundation Server Farm, 410
    - in Team Foundation Server logical architecture, 407
    - virtualization and, 413
  - SQL Tracing, 307, 311
  - staged model of implementation, 564
  - Standish Group 204 report, 522
  - stand-up meetings, 613
  - stateful engines, 202
  - Static Code Analysis tool. *See* code analysis tools
  - static data, 126
  - static test suites, 327
  - step loads, 299
  - stored procedures, 227–230
  - stories. *See* user stories
  - storing load test run data, 308
  - story points, 600–601, 611
  - storyboards, 556
  - StringAssert class, 118
  - structure
    - in profiling applications, 170–171
    - in project management, 527–528
    - of trees, 214–215, 432
    - of T-SQL files, 214–215
  - subject matter experts (SMEs), 559
  - subsystems, 18
  - successful tests, 105–106, 116
  - Summary view, 188–191
  - suppressing messages, 149–151
  - symbol servers, 476–478
  - symbols, debugging, 198
  - synchronous messages, 29–31
  - system calls, 183, 247–250
  - System Info buttons, 365
  - system information, 263
  - System.CodeDom, 158
  - Systems Tab, 559
- T**
- tables views, 310
  - “Take Source Branch Version”, 465
  - target branches
    - in branching plans, 467
    - in Branching Wizard, 457–458
    - in Merging Wizard, 458–459
  - targets, session, 179–180
  - Targets folder, 173, 179
  - task work items, 557
  - TDD (test-driven development), 122
  - team builds, 132, 391
  - team capacity worksheets, 612–613
  - Team Explorer
    - builds in, 479
    - connecting to Team Foundation Server via, 527
    - new team projects in, 526–527
    - portal configuration in, 629
    - product backlog workbooks in, 606
    - product planning workbooks in, 602
    - publishing reports in, 580–583
    - running queries in, 541
    - in Team Foundation Server, 394
    - work items in, 534, 538
  - Team Foundation Administrator group, 616
  - Team Foundation Application Tier, 381
  - Team Foundation Build 2010. *See also* Team Foundation Server 2010
    - agent settings in, 495–496
    - architecture of, 477–478
    - automated tests in, 497–500
    - build controllers in, 475–476
    - Build Defaults section in, 486–487
    - build definitions in, 482–489
    - build details view in, 481–482
    - Build Explorer in, 479–481
    - build notifications in, 476, 491–492
    - build number format in, 497
    - build process in, generally, 492–493
    - Clean Workspace in, 496
    - completed builds in, 480–481
    - configurations in, 495
    - Continuous Integration triggers in, 484
    - customizing in, 500–502
    - DefaultTemplate process in, 493–494
    - deletion options in, 477
    - e-mail alerts in, 492
    - gated check-ins, 475, 485
    - General section in, 482–483
    - Get Version in, 500
    - introduction to, 471–472
    - Lab Management and, 371–374
    - layer diagrams and, 90
    - logging verbosity in, 495

- Team Foundation Build 2010.
  - (*continued*)
  - Manual triggers in, 483
  - new features in, 474–477
  - overview of, 472–473
  - path to publish symbols in, 497
  - private builds in, 475, 491
  - process parameters in, 494–500
  - processes in, 487–488
  - properties exposed for common
    - customizations in, 476
  - queueing builds in, 479–480, 489–491
  - retention policies in, 488–489
  - Rolling Builds in, 484
  - schedules as triggers in, 485
  - symbol and source server
    - integrations in, 476–477
  - Team Explorer in, 479
  - Trigger section in, 483–485
  - Windows Workflow 4.0 in, 474–475
  - workflow activities, adding, 502–504
  - working with, generally, 478–479
  - Workspace section in, 485–486
  - ZIP archives in. *See* ZIP archives
- Team Foundation logical architecture
  - deployment scenarios and. *See* Team Foundation Server 2010, deploying
  - introduction to, 405–408
  - Team Foundation Server application in, 410–411
  - Team Foundation Server Farm in, 410
  - Team Foundation Server instance in, 411–412
  - team project collections in, 408–409
- Team Foundation physical architecture
  - deployment scenarios and. *See* Team Foundation Server 2010, deploying
  - hardware requirements in, 412–413
  - introduction to, 405–406
  - overview of, 412
  - software requirements in, 414
- Team Foundation Power Tools, 492, 622
- Team Foundation Server 2005, 420
- Team Foundation Server 2008
  - architecture of. *See* Team Foundation logical architecture
  - hosted environments in, 418
  - upgrading from, 420
- Team Foundation Server 2010
  - accessing, 392–395, 398–400
  - administration features in, 394, 402
  - adopting, 402–403
  - Application Tier in, 381
  - architecture of. *See* Team Foundation physical architecture
  - Branching Guide, 448
  - branching with. *See* branching builds in. *See* Team Foundation Build 2010
  - command-line tools for, 397–398
  - connecting to project management, 527–528
  - core concepts in, 380
  - defined, 380
  - hosting, 402
  - introduction to, 379
  - Lab Management and, 358
  - merging with. *See* merging Microsoft Office and, 538–544
  - Microsoft Office Excel and, 395–396, 544–547
  - Microsoft Office Project and, 397
  - new features in, 400–401
  - path to instances in, 353
  - process templates in, 385–386, 551, 631
  - project management in. *See* project management
  - publishing test results with, 271
  - reports in. *See* Team Foundation Server Reporting
  - SharePoint and. *See* SharePoint
  - team projects in, 381–385
  - test impact analysis in, 134–136
  - test results on, 335–336
  - Unit Test Framework and, 132
  - version control in. *See* Team Foundation version control
- Windows Explorer integrating with, 399–400
- work item tracking in, 387–388
- Team Foundation Server 2010, deploying. *See also* Team Foundation Server 2010
  - branch folders in, 421
  - enterprises in, 417–418
  - hosted environments in, 418–419
  - individuals in, 416
  - in-place upgrades in, 420
  - introduction to, 415
  - migration upgrades in, 421
  - paths, choosing, 421
  - process templates, upgrading, 421
  - small shops in, 416–417
  - small teams in, 416
  - upgrading from old versions of, 420–421
- Team Foundation Server Administrators Guide, 415
- Team Foundation Server Analysis database, 575–576
- Team Foundation Server application, 410–411
- Team Foundation Server Farm, 410
- Team Foundation Server Installation Guide
  - on Build services, 478
  - on TFS deployment, 414–417, 420–421
- Team Foundation Server instance, 411–412
- Team Foundation Server Reporting. *See also* Team Foundation Server 2010
  - creating reports, 575–576
  - dashboards in, 592, 593–595
  - data warehouse in, 573–574
  - Documents link in, 595
  - Excel Reports in, 595
  - introduction to, 572–575
  - left navigation panes in, 592–595
  - Microsoft Excel Reports in. *See* Microsoft Office Excel Reports
  - OLAP cube in, 574–575
  - operational store in, 573

- out-of-the-box reports in, 589–592
- process guidance links in, 595
- process templates in, 589–592
- project portals in, 592–596
- RDL Reports in, 585–588
- Reports link in, 595
- Team Web Access link in, 593
- top horizontal menus in, 595–596
- working with, generally, 575
- Team Foundation version control
  - branching in, 440–443
  - check-in policies in, 435–436
  - checking in items, 432–434
  - checking out items, 432, 434–435
  - command-line tools in, 444–445
  - histories in, 436–437
  - introduction to, 70, 423–424
  - labeling files in, 437–438
  - merging in, 440, 443–444
  - repositories, adding projects to, 432
  - security roles in, 425–426
  - setting up, 425–427
  - shelving in, 438–439
  - Source Control Explorer in, 427–432
  - Visual SourceSafe 2005 and, 424–425
  - workspaces in, 426–427, 429–432
- team member capacity worksheets, 613
- team project collections. *See also* team projects
  - defined, 523
  - in Team Foundation logical architecture, 408–409
  - in Team Foundation Server, 381–382
- team projects
  - collections. *See* team project collections
  - creating, 523–527
  - per application, 384
  - per release, 384
  - per team, 384–385
  - portals in, 528
  - scope of, 384
  - in Team Foundation Server, 382–385
- Team Queries, 535
- Team System 2010, 244
- Team Tools, 195
- Team Web Access, 593
- templates
  - in Agile Software Development, 562
  - process. *See* process templates
  - T-SQL script, 220–221
  - virtual machine. *See* VM (virtual machine) templates
- Templex, 569
- test agents, 359
- Test and Lab Manager, 559
- test case work items, 559
- test cases
  - in manual tests, 326–327
  - in manual tests with virtual environments, 369
  - in user story work items, 557
- test categories, 267
- Test Configuration Manager, 325
- test impact analysis
  - in automated software tests, 275–276
  - in build process, 498–500
  - code in, 133
  - defined, 263
  - editing files in, 216
  - example of, 133–136
  - introduction to, 131–132
  - limitations of, 275
  - in manual testing, 324–325
  - prerequisites for, 132
  - in software development, 97
  - in test plans, 324–325
  - in Unit Test Framework, 131–136
  - viewing impacted tests, 133
- Test Lab Management, 493
- Test List Editor
  - Test Categories vs., 266
  - in unit tests in Visual Studio Express Edition, 106–107
  - using, 269
- test management in team projects, 386
- Test Management plug-ins, 618
- Test Manager. *See* Microsoft Test Manager
- test metadata files, 266
- test mix model, 300–302
- test plans. *See also* test projects
  - assigning configurations in, 330
  - assigning testers in, 330
  - authoring test cases in, 327–329
  - builds in, 323–324
  - configurations of tests in, 325–326
  - impact analysis in, 324–325
  - introduction to, 320–322
  - for manual testing, 320–322
  - plan contents in, 326–330
  - results of. *See* test results
  - settings for, 322–323
  - shared steps in, 329–330
  - tools for, 264
- test projects
  - for coded user interface testing, 341
  - creating, 265–266
  - defined, 265
  - introduction to, 265
  - planning. *See* test plans
  - results of. *See* test results
  - setting options for, 267
  - for unit testing, 103–104
- test results. *See also* test plans; test projects
  - in automated software tests, 270–271
  - exporting, 270
  - files of, 120
  - filtering, 270
  - grouping, 270
  - introduction to, 269–270
  - publishing, 271
  - sorting, 270
  - unit tests in, 109
  - XML or TRX files and, 270
- test rigs, 313
- Test Runner. *See* Microsoft Test Runner
- test runs, 108–109, 331
- test suites, 326–327
- test tables, 310
- Test View
  - automated software tests in, 267–269
  - executing load tests in, 308
  - unit tests in, 106
- TestClass attribute, 104
- TestCleanup attribute, 111
- TestContext class, 120
- Test-Driven Development: By Example*, 122

*Test-Driven Development in Microsoft.NET*, 122

test-driven development (TDD), 122

testers, 565

test-first experience, 98

testing

automated software. *See* automated software testing

black box, 100

center, 321

coded user interface. *See* coded user interface (UI) testing

databases. *See* database testing

loads. *See* load testing

policy, 435

reports on, 561, 567

settings for, 130–132, 266, 274–275

software. *See* software testing

Web performance. *See* Web performance testing

white box, 100

`TestInitialize` attribute, 111

`TestMethod` attribute, 104–105

`TestMyDatabaseProject.dgen`, 224–225

`TestProperty` attribute, 119–120

`tf.exe`, 444–445

`TFS_Analysis`, 577–578

`TFSBuild.exe`, 490

theory of profound knowledge, 562

think times, 298–299

third-party tools, 102

thread execution, 185

Three-A's, 106

three-tier architecture, 405

Thresholds tables, 311

Tier Interactions property pages, 181

time boxes, 601

toolboxes

in activity diagrams, 25–26

in class diagrams, 48–49

in component diagrams, 36

in sequence diagrams, 29–30

in use case diagrams, 20–21

top horizontal menus

Copy Dashboards in, 596

new Excel report in, 596

new work item link in, 596

in Team Foundation Server

Reporting, 595–596

top-down design

activity diagrams in, 22–27

class diagrams in. *See* class diagrams

component diagrams in. *See* component diagrams

introduction to, 17

sequence diagrams in, 28–31

use case diagrams in, 18–21

Trace and Test Impact test

settings, 274

trace log files, 250

tracing markers, 168

Track Changeset option,

443–444

tracking iterations

in agile planning workbooks, 613–614

introduction to, 613

issues spreadsheets in, 613

retrospectives in, 614

tracking logic, 129

Transact SQL (T-SQL). *See* T-SQL (Transact SQL)

transactions

in database testing, 232

in load testing, 310–311

in Web performance tests, 290

tree lists, 546–547

tree queries, 536

tree structures, 214–215, 432. *See*

*also* branching; merging

Trend Reports, 580–581

triggers

in builds, 483–485

controls for, 391

in database testing, 227–230

TRX files, 265, 313

T-SQL (Transact SQL)

editing files in, 216

file structures, 214–215

script templates in, 220–221

static analysis in, 233–236

types, defined, 47–50, 53

## U

UCLs (Upper Control Limits), 562–563

UI (user interface) experiences, 97

UI (user interface) Layer, 84–85, 87–89

UI (user interface) testing. *See* coded user interface (UI) testing

`UIMap.Designer.cs`, 345, 350

UML (Unified Modeling Language).

*See* Unified Modeling Language (UML)

UNCs (Universal Naming

Conventions), 391

uni-directional dependencies, 87

Unified Modeling Language (UML)

class diagrams in. *See* class diagrams

diagrams in, 11–14

introduction to, 3

in Microsoft's modeling strategy, 5–6

Model Explorer in, 39–42

Unit Test Framework

access in, 122–126

`AssemblyCleanup` attribute, 112–113

`AssemblyInitialize` attribute, 112–113

`Assert` methods in, 113–116

benefits of, 100–101

`ClassCleanup` attribute, 112

`ClassInitialize` attribute, 112

cleanup in, 110–113

code coverage in, 129–131

code generation in, 126–129

`CollectionAssert` class in, 116–118

concepts in, 100–102

custom properties in, 119–120

data-driven unit tests in, 121–122

`ExpectedException` attribute in, 119

initialization in, 110–113

introduction to, 99–100

`PrivateObject` in, 123–125

`PrivateType` in, 125–126

programming with, generally, 110

StringAssert class in, 118  
 test impact analysis in, 131–136  
 TestCleanup attribute, 111  
 TestContext class in, 120  
 TestInitialize attribute, 111  
 TestProperty attribute in,  
 119–120  
 third-party tools in, 102  
 vs. Visual Studio unit tests. *See*  
 unit tests in Visual Studio  
 Express Edition  
 writing effectively in, 101–102  
 unit tests in Visual Studio Express  
 Edition  
 classes of, 104  
 creating, 103–106  
 debugging, 109–110  
 failure of, 105–106  
 identifying, 104–105  
 introduction to, 102–103  
 managing and running, 106–108  
 performance sessions from, 173  
 profiling sessions in, 173  
 running directly from code, 108  
 success of, 105–106  
 Test List Editor in, 106–107  
 Test Results, 109  
 test run configurations, 108–109  
 Test View in, 106  
 Universal Naming Conventions  
 (UNCs), 391  
 unknown content, 579  
 Unreferenced Nodes analyzer, 77  
 upgrading, 420–421  
 uploading process templates, 631  
 Upper Control Limits (UCLs),  
 562–563  
 Urban Turtle Planning, 556  
 URL query parameters, 294  
 Url to launch, 180  
 “Use a secondary data connection to  
 validate unit tests”, 228–229  
 use case diagrams  
 adding activity diagrams to, 27  
 creating, 20–21  
 introduction to, 18  
 in software architecture, 11  
 toolboxes in, 20–21

understanding, 18–20  
 using, 20  
 user gestures, 247  
 user interface (UI) experiences, 97  
 user interface (UI) testing. *See* coded  
 user interface (UI) testing  
 user pace, 301  
 user stories  
 in Excel Reports, 578  
 in iteration planning, 607  
 overview report on, 586  
 in product backlog workbooks,  
 603–604  
 in product planning workbooks, 602  
 progress reports on, 590–591  
 story points in, 600–601  
*User Stories Applied: For Agile  
 Software Development*, 555  
 user story work items, 555–557  
 users for sites, 279  
 using() clause, 349–350  
 utility methods, 272

## V

validation  
 of architecture, 88–90  
 in layer diagrams, 88–90  
 levels, 305  
 in Web performance tests, 291–293  
 verification steps, 328  
 version control. *See* Team  
 Foundation version control  
 VHD (virtual hard disk) files, 402  
 video recorders, 263  
 viewing  
 code coverage results, 130–131  
 distributed load testing, 317  
 impacted tests, 133  
 in Performance Explorer. *See* views  
 in Performance Explorer  
 tests for particular methods, 133  
 views in Performance Explorer  
 Allocation, 193–194  
 Call Tree, 193  
 Caller/Callee, 192–193  
 Functions, 191–192  
 Functions Details, 191–192  
 introduction to, 187–188  
 Objects Lifetime, 194–195  
 Summary, 188–191  
 virtual environments  
 automated build-deploy-test with,  
 371–375  
 creating new test settings in,  
 366–369  
 overview of, 360–366  
 running manual tests with,  
 369–371  
 virtual hard disk (VHD) files, 402  
 virtual machines (VMs). *See* virtual  
 test lab environments  
 virtual test lab environments  
 agents in, 359  
 automated build-deploy-test in,  
 371–375  
 golden images in, 359  
 infrastructure of, 358–359  
 introduction to, 357  
 manual tests with environments in,  
 369–371  
 overview of, 360–366  
 physical environments and, 375  
 test settings in, 366–369  
 testing in, 366–371  
 virtual users, 300  
 virtualization, 259, 413  
 visibility of queries, 537  
 Visual Basic, 62, 341  
 Visual C+, 62  
 Visual SourceSafe (VSS), 388–389,  
 424–425  
 Visual Studio 2010. *See also* Visual  
 Studio 2010 Ultimate  
 branching in. *See* branching  
 coded UI tests in. *See* coded user  
 interface (UI) testing  
 component diagrams in, 37–38  
 generalist testers in, 319  
 IntelliTrace in. *See* IntelliTrace  
 Lab Management in. *See* Lab  
 Management  
 new software development features  
 in, 95–98  
 offline databases development in,  
 201–203

Visual Studio 2010 (*continued*)

Performance Explorer in. *See* Performance Explorer

profiling tools in, 167–169

software testing in. *See* software testing

Team Explorer in, 424, 522–523

Team Foundation Server in. *See* Team Foundation Server 2010

Test Controller Windows, 314

Test Professional in, 260

unit tests in. *See* unit tests in Visual Studio Express Edition

Web performance tests in. *See* Web performance testing

Visual Studio 2010 Premium, 139–140, 260

Visual Studio 2010 Ultimate. *See also* Visual Studio 2010

activity diagrams in, 11–12

Architecture Explorer in. *See* Architecture Explorer

class diagrams in, 13

code analysis tools in, 139–140

component diagrams in, 13

Debug menu, 244

domain-specific languages in, 7

introduction to, 3

layer diagrams in. *See* layer diagrams

new tools in, 10–15

process templates in, 549–550

profiling tools in, 167–169

sequence diagrams in, 12

software testing tools in, 260

use case diagrams in, 11

Visual Studio Express Edition. *See* unit tests in Visual Studio Express Edition

visualizing branching structures, 442

VM (virtual machine) templates, 358, 361–362

VMs (virtual machines). *See* virtual test lab environments

Vorontsov, Alexei A., 122

VSDEBCMD.EXE, 237

VSInstr.exe utility, 184

VSS (Visual SourceSafe), 388–389, 424–425

**W**

WANs (wide area networks), 390

warehouse control services, 573–574

Warm-up duration, 305

warnings, 145–151

watch lists, 253

WCF (Windows Communication Framework), 478

Web applications, 279–281

Web performance testing. *See also* load testing

binding to sources, 294

coded, 294–297

coded user interface tests vs., 278–279

comments in, 290

configuring, 280–282

context parameters in, 290–291

creating, 281–282

data sources in, 293–294

data-driven, 293–294

editing, 289

extraction rules in, 290

introduction to, 277–278

observing, 287–289

parameterizing Web servers for, 285

recording, 282–284

request properties in, 289

running, 287

settings for, 284–287

transactions in, 290

unit tests in, 261

users in, 279

validation rules in, 291–293

Web applications in, 279–281

Web Site Administration Tool, 279

Web Test Editor

data-driven Web performance tests in, 293–294

general Web performance tests in, 284–285

manual Web performance tests in, 281

running tests in, 287–288

Web Test Recorder

coded Web performance tests in, 295

introduction to, 278

overview of, 281–284

Web tests, 279. *See also* Web performance testing

WebTestRequest instances, 295–296

weighting agents, 315

white box testing, 100

whitespace characters warning, 235–236

wide area networks (WANs), 390

wiki libraries, 560, 632

wiki page links, 595

wildcards (\*), 235

Windows

Explorer, 399–400

Hardware Developer Center, 198

Team Foundation Server 2010 and, 414

Workflow 4.0, 474–475

Windows Communication Framework (WCF), 478

Windows Counter property page, 178–179

Windows Events property page, 178

Windows Presentation Foundation (WPF), 98, 340

Windows SharePoint Services (WSS). *See also* SharePoint

document libraries in, 383

Microsoft Office Sharepoint Server vs.. *See* Microsoft Office SharePoint Server (MOSS)

Portal Server, 412–415

Team Foundation Server Administration Console and, 394

upgrading, 420–421

in virtualized environments, 412–413

WIT (work item tracking), 380–381

witadmin command-line utilities, 621

Work Item Add-in, 538, 544

Work Item Categories, 626

Work Item option in Source Control Settings, 435

work item queries, 535–538

work item tracking  
 plug-ins, 618–619  
 in Process Template Editor, 624–626  
 in project management, 523  
 in Team Foundation Server, 380–381, 387–388  
 in team projects, 386  
 work item types (WITs), 529  
 work items  
 in Agile Software Development, 554–560  
 on channel option, 434  
 in CMMI Level 3, 566  
 creating and updating, 534–535  
 link types and, 531–533  
 in process templates. *See* work items in process templates  
 in project management, 529–531  
 in Team Foundation Server, 400–401, 433  
 in team projects, 383  
 tracking. *See* work item tracking  
 XML editors and, 620–621  
 work items in process templates  
 bug, 557–559  
 introduction to, 554–560

issue, 559  
 in Process Template Editor, 629–631  
 shared steps, 560  
 task, 557  
 test case, 559  
 user story, 555–557  
 Work Items node, 527  
 Work Items Picker, 352  
 Workbook Queries, 535  
 workflow activities, 501–504, 513–517  
 workspaces  
 in builds, 485–486  
 in version control, 426–427, 429–432  
 WPF (Windows Presentation Foundation), 98, 340  
 wrapper code, 150  
 (WSS) Windows SharePoint Services. *See* SharePoint  
 www.wrox.com, 340

## X

XAML (Extensible Application Markup Language)  
 builds in, 501–504

process parameters in, 511  
 workflow activities in, 514–516  
 ZIP archives in, 507–509  
 XML (Extensible Markup Language)  
 breakpoints in, 251  
 code analysis results in, 147  
 data tips in, 254  
 data-driven tests in, 347–349  
 data-driven Web performance tests in, 293  
 process templates in, 385, 617–621  
 rule descriptives in, 157  
 Rules.xml in, 161–162  
 Web performance tests in, 281, 294–295  
 XP (eXtreme Programming), 484, 551

## Z

ZIP archives  
 creating, 504–509  
 process arguments in, 509  
 process parameter metadata in, 510–512  
 process parameters in, 509–510  
 workflow activities in, 513–517