

Index

A

Abstract Window Toolkit (AWT)

- about, 3, 4–5
- APIs, 23
- architecture, 4
- components, 4
- impedance matching, 4
- version 1.0, 4
- version 1.1, 4
- widgets, 9

`AbstractDocument`, **409–410**

`AbstractTextEditor`, **412**

`AbstractTreeViewer`, **44**

accessing

- items in combos, 156–158
- items in lists, 163–165
- operating system information, 69
- `TableItems`, 296–297
- text information, 180
- `ToolItems`, 203
- wizard pages, 396–397

accessory resource-based objects, 33

accommodation, changes, 43

Acrobat Reader, 10

action mechanism, `JFace`, 10

actions

- about, 216–219
- creating, 216–219, 480–484
- using, 216–221

adding

- connections, `Draw2D`, 472–473
- controls to toolbar, 207
- data in table, 225–229
- drag-and-drop support, 490–493
- items, combos, 157, 164
- items, lists, 157, 164
- toolbars, 207–210
- wizard pages with `addPages()`, 395

adding filters

- combos, 171
- lists, 171
- tables, 243–244
- trees, 256–267

`addPages`, **395**

adjacent side of another control layouts, 141–143

advanced drawing techniques, 318–320

AIX, 13

appending text, 181

applets, 4

application model data

- creating, 15–16
- `JFace`, 15–16
- SWT, 15–16

Application Program Interfaces (APIs), 9, 32

application window

- contents, 486–489
- creating, 479–480

architecture, AWT, 4

arcs drawing, 311–313

`asyncExec`, 74–75

attaching

- adjacent side of another control layouts, 141–143
- control position in parent composite layouts, 139–141
- control to another control layouts, 141–143

automatically parsing URLs, eclipse forms, 436–437

B

background color, `TableItem`, 226–228

bar menus, 193–194

bitmap files, 11

bmp files, 11

Borland, 8

browsers, 298–305

buffering, double, 320

building

- combos, 154–156
- lists and capturing item selections, 160–163
- skeleton application, 479–489
- tree up with MVC approach, 17–18
- tree up with traditional approach, 16–17

Button, **4, 20, 27**

button1, **33**

button2, **33**

Button peer widget, **21**

ButtonPeer, **20–21**

C

C++, **18**

cancel processing with `performCancel()`, **396**

Canvas, **4, 310**

capturing

- events, 168–171
- events with Draw2D, 473–474
- item selections, 154–156
- modification events, 183

CCombo class, **159**

cell editors, setting, **239–242**

centering control on another control, **143**

changes, accommodation, **43**

Checkbox, **4**

CheckboxTableViewer, **45**

CheckboxTreeViewer, **45**

children

- disposing of parent disposes, 32–33
- laying out of composite, 121–122
- resizing SashForm, 292–293

Choice, **4**

clipboard

- about, 331
- adding drag-and-drop support, 345–347
- bookmark organizer, 338–347
- constructing UI, 339–343
- creating custom transfer type, 343–345
- creating drag sources, 332–335
- creating drop targets, 335–338
- `dragAccept()`, 338
- `dragEnter()`, 337
- `dragFinished()`, 335
- `dragLeave()`, 338
- `dragOperationChanged()`, 338
- `dragOver()`, 338
- `dragSetData()`, 335

`dragStart()`, 334

`drop()`, 338

`DropTargetListener`, 336–337

getting data, 348–349

putting data on, 347–348

using, 347–349

using drag and drop, 331–347

clipping, SWT, **318–319**

coding SWT program, **51–53**

ColdFusion **MX**, **18**

Color, **28, 31**

color, background, **226–228**

ColorDialogs, **262–266**

column-wise sorting using sorters, **242–243**

ColumnLayouts, **442–443**

combos

- about, 153
- accessing items, 156–158, 163–165
- adding filters, 171
- adding items, 157, 164
- building and capturing item selections, 154–156
- building lists and capturing item selections, 160–163
- capturing events and getting selections, 168–171
- CCombo class, 159
- creating domain-specific model objects, 165–166
- creating ListViewer, 166–167
- creating with sorted list, 158–159
- finding items, 156–157, 164
- getting items, 156, 163
- lists, 154–159
- removing items, 157–158, 164–165
- setting content provider and content, 167
- setting items, 156, 163
- setting label provider, 168
- setting sorter, 172
- single and multi selection lists, 159
- with sorted list, 158–159
- styles, 153–154
- updating/refreshing viewer, 172–175
- using, 154–159
- using lists, 159–165
- using ListViewers, 165–175

components, JFace, **10**

composite

- attaching control position in parent composite layouts, 139–141
- children, laying out of composite, 121–122

Composite
 about, 32
 class, 98–103
 focus traversal, 99–102
 layout, 99
 styles, 98–99
 composite1, **34–36**
 composite2, **34**
configuring
 Eclipse, 48–50
 IDEs, 49–51
 JBuilder, 50–51
connections, drawing diagrams with Draw2D,
 adding, 472–473
construct toolbar, 485
 Container, **4**
content assistance, providing, 418–422
content provider
 about, 18
 setting, 238
 setting combos, 167
 setting trees, 254–255
control position in parent composite layouts,
 attaching, 139–141
controller-related classes, 410–412
controllers
 reuse, 43
 text editor creation with JFace Text, 409–412
controls
 adding to toolbar, 207
 attaching to another, 141–143
 Controls, **36**
 CoolBars, **210–212**
copying images, 314–316
 createControl, **400**
custom editor, creation with JFace Text, 412–415
custom form controls, 432–440
customizing
 eclipse forms, 431–432
 TabItems, 297

D

data in table, adding, 225–229
debugging
 dispose, 30
 SWT, 27
 DefaultPartitioner, **410**

dialogs
 about, 261–262
 ColorDialogs, 262–266
 creating, 272–274
 DialogSettings, 403
 DirectoryDialogs, 266–270
 file selection dialog box, 5
 file selection dialog user interfaces, 6–7
 FileDialogs, 266–270
 FontDialogs, 262–266
 IDialogSettings, 402
 InputDialogs, 384–385
 JFace, 10, 382–386
 JFace wizards, 402–404
 loading and saving dialog settings, 402–404
 MessageBoxes, 270–271
 MessageDialogs, 382–384
 ProgressMonitorDialogs, 385–386
 WizardDialog, 391–392, 401
 DialogSettings, **402, 403**
direct palettes, 322–323
 DirectoryDialogs, **266–270**
 Display, **28, 37**
displaying animation, 327–329
displays
 accessing operating system information, 69
 asyncExec, 74–75
 event handling fundamentals, 67–71
 event handling with displays, 69–71
 event model, 77–87
 events, listeners, and listener notification process,
 78–87
 multithreaded UI programming, 71–77
 multithreading with UI thread and non-UI threads,
 71–77
 native event handling mechanism, 67–71
 SWT event handling, 68
 SWT time-consuming operation UI pattern, 74–77
 syncExec, 75–76
 thread-safe UI calls, 74–77
 timerExec, 76–77
 typed events and typed event listeners, 84–87
 untyped events and untyped event listeners, 81–84
 use, 74–76
 dispose event, **31**
 dispose() **method, 28–31**
disposed listener, 31
DisposeListener, 34

disposing

- how to, 30
- if did not create, 31–32
- of parent, 32–33
- shell, 13–14
- when to, 30–31

Document, **409–410**

domain-specific model objects

- combos, 165–166
- creating, 236–237
- lists, 165–166

double buffering, 320

downloading SWT/JFace, 47–48

drag and drop

- about, 331
- adding, 345–347, 490–493
- bookmark organizer, 338–347
- constructing UI, 339–343
- creating custom transfer type, 343–345
- creating drag sources, 332–335
- creating drop targets, 335–338
- `dragAccept()`, 338
- `dragEnter()`, 337
- `dragFinished()`, 335
- `dragLeave()`, 338
- `dragOperationChanged()`, 338
- `dragOver()`, 338
- `dragSetData()`, 335
- `dragStart()`, 334
- `drop()`, 338
- getting data from clipboard, 348–349
- overview of methods in `DropTargetListener`, 336–337
- putting data on clipboard, 347–348
- using clipboard, 347–349
- using drag and drop, 331–347

drag source, 331

drawing

- arcs, 311–313
- graphics contexts, 307–321
- images, 314–316
- lines, arcs, shapes, 311–313
- shapes, 311–313
- text, 316–318

drawing diagrams with Draw2D

- about, 463–465
- adding connections, 472–473
- capturing events, 473–474
- creating simple UML diagrams, 465–472

`drawRectangle` **method, 29**

drop-down menus, 194

drop target, 331

E

Eclipse Forms

- about, 427–428
- automatically parsing URLs, 436–437
- basic form, 428–431
- basic hyperlinks, 432–433
- customizing, 431–432
- `ImageHyperlinks`, 433–434
- managing multiple hyperlinks with `HyperlinkGroups`, 434–435
- render XML text, 437–438
- rendering normal text, 435–436
- using `ColumnLayouts`, 442–443
- using custom form controls, 432–440
- using `ExpandableComposites`, 438–439
- using form layout managers, 440–443
- using `FormTexts`, 435–438
- using `Hyperlinks`, 432–435
- using sections, 439–440
- using `TableWrapLayout`, 441–442

Eclipse platform, 8

Embarcadero, 18

embedding Microsoft Word OLE Document into SWT application, 447–454

event handling

- about, 68
- accessing operating system information, 69
- `asyncExec`, 74–75
- with displays, 69–71
- event model, 77–87
- events, listeners, and listener notification process, 78–87
- fundamentals, 67–71
- listener, 4
- listeners, 78–87
- model, 77–87
- multithreaded UI programming, 71–77
- multithreading with UI thread and non-UI threads, 71–77
- native event handling mechanism, 67–71
- source, 4
- SWT components, 299–301
- SWT event handling, 68
- SWT time-consuming operation UI pattern, 74–77
- `syncExec`, 75–76
- thread-safe UI calls, 74–77

timerExec, 76–77
 typed events and typed event listeners, 84–87
 untyped events and untyped event listeners, 81–84
 use, 74–76
 using displays, 68
 wake, 77
 ExpandableComposite, **438–439**

F

file chooser

Swing, 12
 SWT, 11–12

file selection dialog box, 5

file selection dialog user interfaces, 6–7

FileDialogs, **266–270**

files

bitmap, 11
 identifying files for deployment, 61
 packaging and signing files, 61–62

filling shapes, 313–314

FillLayouts, **122–124**

filters

adding combos, 171
 adding lists, 171
 adding tables, 243–244
 adding trees, 256–267

filters, tables, 243–244

finalize method, 29

finding items, 156–157, 164

flexibility

JFace, 14–18
 SWT, 14–18

focus traversal, 99–102

Font, **28**

FontData, **37, 38**

FontDialogs, **262–266**

fonts

FontRegistry, 36–38
 registries, 10

foreground colors for cells in TabItem, 226–228

form layout managers, 440–443

FormLayouts, **138–144**

forms, Eclipse Forms, 428–431

FormTexts, **435–438**

Fortran, 18

FrontPage class, **400–401**

Fujitsu, 8

G

garbage collection mechanism, 28

gc, **28, 29**

getPageCount, **396**

getting items, 156, 163

Gosling, James, 7

graphics

context, 308
 drawable object, 309
 SWT, 308–309

GridData, **128–133**

GridLayouts, **127–128**

groups, SWT, **287–290**

H

handling

events, 195–196
 events, trees, 249–250
 selections, tables, 229–230

horizontal alignment layouts, 128–130

horizontal indentation layouts, 130

horizontal span layouts, 130–131

HP, 8

HP-UX, 13

HyperlinkGroups, **434–435**

hyperlinks, Eclipse Forms, **432–435**

I

IBM, 8

identifying files for deployment, 61

IDialogSettings, **402**

IDocument, **409–410**

IDocumentPartitioner, **410**

IDocumentProvider, **412**

Image, **28**

ImageData, **322–324**

ImageDescriptor, **37**

ImageHyperlinks, **433–434**

ImageRegistry, **37, 38–41**

images

copying, 314–316
 handling, SWT, 321–329
 registries, 10
 scaling, SWT, 327
 SWT, 321–322

impedance matching, 4

implementation

SWT, 27–28

table viewers, 489–490

indexed palettes, 324

inserting text, 181

installing SWT/JFace, 47–48

Intel, 18

IPartitionTokenScanner, **410**

ISourceViewer, **411–412**

isPageCompleted, **401**

items, combos

about, 156, 163

adding, 157, 164

setting, 156, 163

ITextView, **410**

ITreeContentProvider, **18**

IWizard, **391**

IWizardContainer, **391**

J

jar, **40**

Java

early programs, 4

garbage collection mechanism, 28

version 1.2, 5

Java Foundation classes (JFC), 5–8

Java Language Specification (JLS), 29–30

Java Native Interface (JNI), 19

Java user interface (UI) toolkits

about, 3

Abstract Window Toolkit (AWT), 3, 4–5

evolution of GUI frameworks, 3–10

JFace, 10–18

Standard Widget Toolkit (SWT), 8–10

Swing, 5–8

Java Web Start, 60–66

java.awt.Button **class**, **19, 20**

JavaSoft, 5

javax.swing, **6**

javax.swing.border, **6**

javax.swing.event, **6**

javax.swing.plaf, **6**

javax.swing.table, **6**

javax.swing.text, **6**

javax.swing.tree, **6**

javax.swing.undo, **6**

JFace

about, 3, 10

action mechanism, 10

building tree up with MVC approach, 17–18

building tree up with traditional approach, 16–17

components, 10

creating application model data, 15–16

dialogs and wizards, 10

flexibility, 14–18

image and font registries, 10

maturity, 18

mechanisms, 44–45

model-view-controller (MVC) approach, 14

progress reporting for long-running operations, 10

traditional programming approach, 14

viewers and editors, 10

JFace dialogs

about, 373

application Windows, 374–378

InputDialog, 384–385

JFace dialogs, 382–386

JFace windows, 373–382

MessageDialogs, 382–384

multiple windows management with WindowManagers,
381–382

org.eclipse.jface.window.Window, 373–374

ProgressMonitorDialogs, 385–386

running time-consuming operations with application
windows, 378–381

JFace Text

AbstractDocument, 409–410

AbstractTextEditor, 412

controller-related classes, 410–412

controllers, 409–412

creating custom editor, 412–415

DefaultPartitioner, 410

Document, 409–410

framework, 407–412

IDocument, 409–410

IDocumentPartitioner, 410

IDocumentProvider, 412

IPartitionTokenScanner, 410

ISourceViewer, 411–412

ITextView, TextView, 410

model-related classes, 409–410

models, 409–412

package organization, 407–408

providing content assistance, 418–422

running editor, 423–424

SourceViewer, 411–412

SourceViewerConfiguration, 411–412

SWT/JFace Mechanisms, 45

syntax highlighting, 415–418

TextEditor, 412
views, 409–412

JFace viewers, 44–45

JFace windows

about, 373
application Windows, 374–378
InputDialog, 384–385
JFace dialogs, 382–386
JFace windows, 373–382
MessageDialogs, 382–384
multiple windows management with WindowManagers, 381–382
org.eclipse.jface.window.Window, 373–374
ProgressMonitorDialogs, 385–386
running time-consuming operations, 378–381

JFace wizards

about, 391–393
accessing Wizard pages, 396–397
adding wizard pages with addPages(), 395
cancel processing with performCancel(), 396
creating Wizard Pages, 397–401
finish processing with performFinish(), 395–396
loading and saving dialog settings, 402–404
running, 401

L

Label, 4

label provider

about, 18
setting, 238–239
setting combos, 168
setting trees, 255

LabelProvider, **18**

layouts

about, 119
attaching control position in parent composite, 139–141
attaching control to another control, 141–143
attaching to adjacent side of another control, 141–143
centering control on another control, 143
children of composite, 121–122
composite, 99
creating, 145–150
data objects, 121
FillLayouts, 122–124
FormLayouts, 138–144
grabbing excess space, 131–133
GridData Objects, 128–133
GridLayouts, 127–128, 127–138

horizontal alignment and vertical alignment, 128–130
horizontal indentation, 130
horizontal span and vertical span, 130–131
introduction, 119–122
laying out children of composite, 121–122
layout data objects, 121
object, 33
RowData Objects, 126
RowLayouts, 124–126
sample GUI using GridLayouts, 133–138
setting, 120–122
size, 133
StackLayouts, 144–145
terminology, 119–120
LineBackgroundListeners, **190–191**

lines

drawing, 311–313
setting backgrounds, 187–188

LineStyleListeners, **188–190**

Linux, 13

List, 4, 42

listener event, 4

lists

about, 153
accessing items in combos, 156–158
accessing items in lists, 163–165
adding filters, 171
adding items, 157, 164
building combos and capturing item selections, 154–156
building lists and capturing item selections, 160–163
capturing events and getting selections, 168–171
CCombo class, 159
combo styles, 153–154
combos, 159–165
creating combos with sorted list, 158–159
creating domain-specific model objects, 165–166
creating ListView, 166–167
finding items, 156–157, 164
getting items, 156, 163
ListViewers, 165–175
removing items, 157–158, 164–165
setting content provider and content, 167
setting items, 156, 163
setting label provider, 168
setting sorter, 172
single and multi selection lists, 159
updating/refreshing viewer, 172–175
using, 159–165
using combos, 154–159

ListViewer, **45, 165–175**
loading dialog settings, **402–404**
low-level widgets, **28**

M

Mac OS X, 13
Macromedia, 18
managing multiple hyperlinks with HyperlinkGroups, 434–435
maturity
 JFace, 18
 SWT, 18
maximizing control in SashForm, 293–294
Menu, 32, 34, 36
menu bar, 484
MenuItem, **34, 196–198**
MenuManagers, **219–220**
menus
 about, 193
 accessing ToolItems, 203
 adding controls, 207
 adding toolbars, 207–210
 bar, 193–194
 creating actions, 216–219
 creating CoolBar with CoolItems, 210–212
 creating menus with MenuManagers, 219–220
 creating text editor, 199–202
 creating toolbars with ToolBarManagers, 220–221
 creating ToolItems, 204–207
 drop-down, 194
 drop-down menus, 194
 handling events, 195–196
 item management, 195
 pop-up, 194–195
 saving and loading CoolBar, 212–215
 shared menus side effect, 34–36
 styles of ToolItems, 204
 ToolBar styles, 202–203
 using, 193–196
 using actions and ContributionManagers, 216–221
 using CoolBars and CoolItems, 210–215
 using MenuItem, 196–198
 using ToolBars, 202–203
 using ToolItems, 203–207
MessageBoxes, **270–271**
Microsoft Windows, 5
Microsoft Word, 10
model-related classes, text editor creation, 409–410

model-view-controller (MVC)
 about, 14, 41
 architecture, 42–44
 benefits, 43
 controller, 42
 costs, 43
 interaction, 42–43
 model, 42
 view, 42
models
 model-view-controller (MVC), 42
 text editor creation with JFace Text, 409–412
Motif toolkit, 25–27
multiple-line text controls, 179–180
multiple views, MVC, 43
multithreaded UI programming, 71–77

N

native event handling mechanism, SWT, 67–71
navigation methods, SWT, 298

O

Object Technology International (OTI), 8
OLE automation, 545–561
OLE container, 447–448
OLE site, 448–449
OleClientSite, **446**
OleControlSite, **446**
OleFrame, **446**
operating system information, accessing, 69
operating system resources
 Swing, 29
 SWT/JFace Mechanisms, 28–29
Oracle, 8
org.eclipse.swt.custom, **28**
org.eclipse.swt.graphics, **28**
org.eclipse.swt.widgets, **28**
org.eclipse.swt.widgets.Button, **19, 23–25**
OS/2, 11
OS.SetWindowText, **27**
OS.XtSetValues, **27**

P

packages, Swing, 6
packaging and signing files, 61–62
PaletteData, **322–324**
Panel, 4

parentWidget, 33
pasting text, 181
performCancel, 396
performFinish, 395–396
photo uploader program
 shell, 13–14
 Swing, 6
 SWT, 9, 11
platforms
 Eclipse, 8
 SWT widget classes, 23
pop-up menus, 194–195
portability, SWT, 13
preparation, starting with SWT/JFace, 47–51
Printer, 28
Printers, 30
printing
 about, 351
 basic, 355–366
 fundamentals, 351–353
 image viewer application, 356
 images, 360–362
 print preview, 362–366
 selecting target printer, 353–355
 setting page margins, 357–360
 text printing and pagination, 366–371
programming OLE in Windows
 about, 445
 activating OLE object, 450
 creating OLE container, 447–448
 creating OLE site, 448–449
 deactivating OLE object, 451
 embedding Microsoft Word OLE Document into SWT
 application, 447–454
 executing common commands, 453–454
 getting and setting property values, 459–460
 invoking methods, 460–461
 listing OLE automation properties and methods,
 454–459
 OLE automation, 545–561
 OleClientSite, OleControlSite, 446
 OleFrame, 446
 saving changes, 452
programming SWT, 13–14
progress reporting for long-running operations, JFace,
 10
ProgressBars, 282–286
property values, 459–460
public Font get, 37

public void, 32
public void put, 37

Q

QNX, 13

R

Rational Software, 18

Red Hat, 8

Region, 28
 Region object, 32
removing items
 combos, 157–158, 164–165
 lists, 157–158, 164–165
removing TabItems, 228–229
rendering
 normal text, 435–436
 XML text, 437–438
 ReservationWizard class, 396
resizing SashForm's children, 292–293
resource management
 SWT, 13
 SWT/JFace, 28–41
resources, disposing if you didn't create it, 31–32
rewriting temperature converter, 58–60
 RowData Objects, 126
 RowLayout, 124–127
 RowLayouts, 124–126
running
 JFace Wizards, 401
 SWT/JFace, 64–66
 SWT program, 53–54
 text editor, 423–424

S

sample application

about, 477–478
 adding drag-and-drop support, 490–493
 building skeleton, 479–489
 construct toolbar, 485
 creating actions, 480–484
 creating application window, 479–480
 creating application window contents, 486–489
 creating menu bar, 484
 implementing table viewers, 489–490
sample GUI using GridLayouts, 133–138
SAP, 8

sashes, 290–291

SashForm, 290–294

saving

- changes, OLE in Windows, 452
- coolbars, 212–215
- dialog settings, 402–404

scales

- about, 277
- using ProgressBars, 282–286
- using scales, 277–280
- using sliders, 280–282

Scrollbar, 4

ScrollPane, 4

sections, Eclipse Forms, 439–440

selecting

- TableItems, 296–297
- trees, 257–259

SelectionListener, **34, 39**

setDates, **400**

setErrorMessage, **400**

setLabel method, **19–20**

setText, **19**

setting

- background and foreground colors for cells in
TableItem, 226–228
- cell editors, Tables, 239–242
- content provider, 238
- content provider, trees, 254–255
- content provider and content, combos, 167
- items, combos, 156, 163
- items, lists, 156, 163
- label provider, 238–239
- label provider, combos, 168
- label provider, trees, 255
- layouts, 120–122
- line backgrounds, text controls, 187–188
- text labels and image labels for cells in a TableItem,
226
- text styles with StyleRanges, 184–187

setting sorter

- combos, 172
- lists, 172
- trees, 256

SetWindowTextW, **25**

shapes

- drawing, 311–313
- filling, 313–314

shared menus side effect, 34–36

shell

- disposing of shell, 13–14
- photo uploader program, 13–14

shell.dispose(), **33**

single-line text controls, 178–179

single lists, 159

size layouts, 133

sliders, 280–282

Solaris 8, 13

sorting table by column, 233–235

source event, 4

SourceViewer, **411–412**

SourceViewerConfiguration, **411–412**

space layouts, 131–133

speed

- Swing, 12–13
- SWT, 12–13

StackLayouts, **144–145**

Standard Widget Toolkit (SWT)

- about, 3, 8
- advantages, 10, 11–18
- APIs, 9
- building tree up with MVC approach, 17–18
- building tree up with traditional approach, 16–17
- creating application model data, 15–16
- debugging, 27
- file chooser, 11–12
- flexibility, 14–18
- implementation, 27–28
- Java user interface (UI) toolkits, 8–10
- low-level widgets, 28
- maturity, 18
- model-view-controller (MVC) approach, 14
- portability, 13
- programming, 13–14
- resource management, 13
- speed, 12–13
- support for native window systems, 11–12
- traditional programming approach, 14
- widgets, 9

Standard Widget Toolkit (SWT)/JFace mechanisms

- did not create, 31–32
- disposing of parent disposes children, 32–33
- disposing of resources, 29–32
- implementation, 19–28
- Java Native Interface (JNI), 19
- JFace and MVC, 44–45
- JFace Text, 45
- JFace viewers, 44–45

- managing fonts, 36–41
- managing images with JFace, 36–41
- model-view-controller (MVC) pattern, 41–44
- MVC architecture, 42–44
- MVC benefits, 43
- MVC controller, 42
- MVC costs, 43
- MVC interaction, 42–43
- MVC model, 42
- MVC view, 42
- operating system resource management rules, 29–36
- operating system resources, 28–29
- resource management with SWT/JFace, 28–41
- shared menus side effect, 34–35
- UI delegation, 44
- using FontRegistry, 36–38
- using ImageRegistry, 38–41
- When to dispose, 30–31

Standard Widget Toolkit (SWT) components

- about, 287
- accessing and selecting TabItems, 296–297
- browsers, 298–305
- customizing TabItems, 297
- events, 299–301
- groups, 287–290
- maximizing control in SashForm, 293–294
- navigation methods, 298
- resizing a SashForm’s children, 292–293
- sashes, 290–291
- sashes and SashForms, 290–294
- SashForms, 291–292
- TabFolders and TabItems, 294–298
- usage, 294–296
- Web browser, 301–305

Standard Widget Toolkit (SWT) graphics and image handling

- about, 307
- advanced drawing techniques, 318–320
- clipping, 318–319
- creating graphics context instance on drawable object, 309
- direct palettes, 322–323
- displaying animation, 327–329
- double buffering, 320
- drawing and copying Images, 314–316
- drawing lines, arcs, shapes, 311–313
- drawing text, 316–318
- drawing with graphics contexts, 307–321

- filling shapes, 313–314
- getting graphics context, 308–309
- image basics, 321–322
- image dandling, 321–329
- image scaling, 327
- ImageData and PaletteData, 322–324
- Indexed Palettes, 324
- obtaining existing graphics context, 308
- transparency and alpha blending, 325–327
- using canvas, 310
- XOR, 319

Standard Widget Toolkit (SWT) program

- coding, 51–53
- starting with SWT/JFace, 51–60

Standard Widget Toolkit (SWT) widgets

- buttons, 112–114
- CLabel class, 116–117
- composite class, 98–103
- composite class focus traversal, 99–102
- composite class layout, 99
- composite class styles, 98–99
- constructor public widget, 90–93
- control class, 93–98
- control methods, 98
- control size and location, 94–95
- control styles, 93–96
- creating irregular-shaped shells, 108–110
- creating shells, 104–107
- dispose of widget, 92
- event handling, 92–93
- fonts and colors, 95
- getting and setting application data, 93
- label basics, 116–117
- label styles, 115
- labels, 115–117
- public display, 91–92
- public void, 92–93
- radio buttons, 113–115
- setting images, 110
- setting title, 110
- shell default button, 110–111
- shell events, 107–108
- shell states, 104
- shell styles, 103
- shells, 103–111
- SWT/JFace UI component hierarchy, 89–90
- UI updating, 95–97
- widget class, 90–93

starting with SWT/JFace

- about, 47
- coding SWT program, 51–53
- configure Eclipse, 48–50
- configure JBuilder, 50–51
- configuring IDEs, 49–51
- downloading and installing SWT/JFace, 47–48
- identifying files for deployment, 61
- Java Web Start, 60–66
- packaging and signing files, 61–62
- preparation, 47–51
- rewriting temperature converter, 58–60
- running SWT program, 53–54
- software deployment with Java Web Start, 60–66
- SWT program, 51–60
- temperature converter application, 55–57
- uploading and running, 64–66
- writing JNLP script, 63–64

structured/viewer, 44

StyledTexts, **184–188**

styles

- composite, 98–99
- ToolItems, 204

Sun Microsystems, 4, 5

sun.awt.windows.WButtonPeer, **21–22**

support for native window systems, Standard Widget

Toolkit (SWT), **11–12**

Swing

- about, 3, 5
- disadvantages, 7
- file chooser, 12
- file selection dialog user interfaces, 6–7
- operating system resources, 29
- packages, 6
- photo uploader program, 6
- speed, 12–13

Sybase, 8

syncExec, **75–76**

syntax highlighting, text editor, 415–418

T

TabFolders, **294–298**

TabItems

- about, 40, 296–297
- background color for cells, 226–228
- creating, 225–226

Table, 32

TableColumns, **224–225**

TableEditors, **230–233**

tables

- adding data in table, 225–229
- adding filter, 243–244
- column-wise sorting, 242–243
- creating, 223–224
- creating TableViewer, 237–238
- creating domain-Specific Model Objects, 236–237
- creating TableItems, 225–226
- defining Table columns, 224–225
- handling selections, 229–230
- removing TableItems, 228–229
- setting background and foreground colors for cells, 226–228
- setting cell editors, 239–242
- setting text labels and image labels for cells, 226
- setting content provider, 238
- setting Label Provider, 238–239
- sorting a table by column, 233–235
- using TableEditors, 230–233
- using Tables, 223–235
- using TableViewers, 235–244

TableTreeViewer, **45**

TableViewer, **45, 235–244**

TableWrapLayout, **441–442**

temperature converter application, 55–57

text

- about, 177
- accessing Text Information, 180
- appending, pasting, inserting text, 181
- capturing modification events, 183
- controls, 178–180
- creating multiple-line text controls, 179–180
- creating single-line text controls, 178–179
- drawing, 316–318
- setting Line backgrounds, 187–188
- setting text styles with StyleRanges, 184–187
- StyleRanges, 184–187
- styles, 177–178
- text basics, 178–180
- text operations, 181–183
- text selections, 183–184
- text styles, 177–178
- using LineBackgroundListeners, 190–191
- using LineStyleListeners, 188–190
- using LineStyleListeners and LineBackgroundListeners, 188–191
- using StyledTexts, 184–188
- using texts, 177–191
- verifying Input, 181–183

text editor

- about, 199–202
- AbstractTextEditor, TextEditor, IDocumentProvider, 412
- controller-Related classes, 410–412
- creating custom editor, 412–415
- IDocument, AbstractDocument, Document, 409–410
- IDocumentPartitioner, IPartitionTokenScanner, DefaultPartitioner, 410
- ISourceViewer, SourceViewer, SourceViewerConfiguration, 411–412
- ITextViewer, TextViewer, 410
- JFace Text Package Organization, 407–408
- model-related classes, 409–410
- models, views, and controllers in JFace Text Framework, 409–412
- overview of JFace Text Framework, 407–412
- providing content assistance, 418–422
- running editor, 423–424
- syntax highlighting, 415–418

TextArea, **4**

TextComponent, **4**

TextEditor, **412**

TextField, **4**

texts, 177–191

TextViewer, **410**

threading

- accessing operating system information, 69
- asyncExec, 74–75
- event handling fundamentals, 67–71
- event handling with displays, 69–71
- event model, 77–87
- events, listeners, and listener notification process, 78–87
- multithreaded UI programming, 71–77
- multithreading with UI thread and non-UI threads, 71–77
- native event handling mechanism, 67–71
- SWT event handling, 68
- SWT time-consuming operation UI pattern, 74–77
- syncExec, 75–76
- thread-safe UI calls, 74–77
- timerExec, 76–77
- typed events and typed event listeners, 84–87
- untyped events and untyped event listeners, 81–84
- use, 74–76
- using displays, 68
- wake, 77

timerExec, **76–77**

ToolBarManagers, **220–221**

toolbars

- accessing toolItems, 203
- adding, 207–210
- adding controls to, 207
- adding toolbar for text editor, 207–210
- bar menus, 193–194
- construct toolbar, 485
- creating actions, 216–219
- creating CoolBar with CoolItems, 210–212
- creating menus with MenuManagers, 219–220
- creating text Editor, 199–202
- creating toolbars with ToolBarManagers, 220–221
- creating ToolItems, 204–207
- drop-down menus, 194
- handling events, 195–196
- menu item management, 195
- pop-up menus, 194–195
- saving and loading display CoolBar, 212–215
- styles, 202–203
- styles of toolItems, 204
- using, 202–203
- using actions and ContributionManagers, 216–221
- using CoolBars and CoolItems, 210–215
- using MenuItemItems, 196–198
- using menus, 193–196
- using menus and menu items, 193–202
- using ToolItems, 203–207

ToolItems, **204–207**

traditional programming approach, SWT, 14

transparency and alpha blending, 325–327

tree, 32

tree widget, 9

TreeEditors, 251–253

TreelItem, 32

TreeItems, **246–249**

trees

- about, 245
- adding filters, 256–267
- creating, 245–246
- creating TreeViewer, 254
- getting selections, 257–259
- handling events, 249–250
- setting content provider, 254–255
- setting label provider, 255
- setting sorter, 256
- using TreeEditors, 251–253
- using TreelItems, 246–249
- using trees, 245–253
- using TreeViewer, 253–259

TreeViewer

TreeViewer, **45, 253–259**
typed event listeners, 84–87
typed events, 84–87

U

UI delegation, SWT/JFace Mechanisms, 44
UML diagrams with Draw2D, creating, 465–472
UML modeling tool, 18
untyped event listeners, 81–84
untyped events, 81–84
updating/refreshing viewer
 combos, 172–175
 lists, 172–175
Upload button, 14
uploading SWT/JFace, 64–66

V

verifying input, 181–183
vertical alignment layouts, 128–130
vertical span layouts, 130–131
Viewer, 44
viewers and editors, JFace, 10
views, text editor creation, 409–412
Visual C++, 11

W

wake, **77**
WebSphere Application Studio, 18
WidgetDisposed, **31**
widgets
 about, 28
 AWT, 9
 low-level, 28
 Standard Widget Toolkit (SWT), 9
widgetSelect **method, 34**
windows, JFace
 about, 373
 application Windows, 374–378

InputDialog, 384–385
JFace dialogs, 382–386
JFace windows, 373–382
MessageDialogs, 382–384
 multiple windows management with WindowManagers,
 381–382
 org.eclipse.jface.window, 373–374
 ProgressMonitorDialogs, 385–386
 running time-consuming operations, 378–381

Windows 98, 13
Windows 2000, 13
Windows ActiveX controls, 10
Windows CE, 13
Windows file chooser, 11
Windows ME, 13
Windows XP, 13
wizard container, 391
wizard pages, 395
WizardDialog, **391–392, 401**
WizardPage, **401**
wizards, JFace
 about, 10, 391–393
 accessing wizard pages, 396–397
 adding wizard pages, 395
 cancel processing, 396
 creating, 393–397
 creating wizard pages, 397–401
 finish processing, 395–396
 loading and saving dialog settings, 402–404
 running, 401
writing JNLP script, 63–64

X

XDE Professional, 18
XOR, 319
XtSetValues, **27**