Index

A

abbreviated filenames, 44
Account buttons, 66
Acquire global variables, 105, 109
acquisition section (requirements specifications), 26
algorithms, labels, 305
Align Objects menu, 50, 94
aligning
  dialog VI panels, 51
  objects, 50
Although Generate Report VI, 171
AML (Assistance Markup Language), 312
analysis section (requirements specifications), 26
Analyze Data events, 267
applications
  Capacitor Test & Sort, 79
  data types, 165-169
  dialogs, 51-52
  enums, 306
  frameworks, 239, 272
    Dynamic Framework, 272-278
    examples, 287-297
    Modular Multiple-Loop Framework, 283-286
    Multiple-Loop Framework, 278-282
  Left to Right VI, 121-122
  online documentation, 314
  optical filter test, 127-128
  performance, 17
  Right to Left VI, 120
  Spectralyzer, 82-83
applying
  controls, 52
  dialogs, 51-52
  fonts, 62
arrays
  descriptions, 301
  simple, 181-184
assigning
  error clusters, 144
  references, 144
  terminals, 145-154

Assistance Markup Language (AML), 312
automatic error handling, 230
avoiding overlapping objects, 57-58

B

background icons, 135
backups, LabVIEW.ini file, 33
Backwards Wire failures, 323-325
banners, 148
best practice requirement specifications, 24-25
binary notation, 176
black borders
  applying, 132
  selecting, 135
Blank state, 256
block diagrams, 87
  Centrifuge DAQ VIs, 122-123
  data flow, 101-111, 114
  documentation, 304-308
  Excessively Nested VIs, 117
  Haphazard VIs, 119-120
  LabVIEW environment options, 32
  layout, 88-93
  maintainability, 11
  readability, 9-11
  Screw Inspection VIs, 124-126
  wiring, 95-101
Bloomy Controls, 147
Boolean data types, 173-174
borders, applying, 132
Bottom Edges alignment tools, 95
Browse buttons, 180
bugs. *See* error handling
Bundle By Name function, 183, 308
business objectives, 22
buttons
  Account, 66
  Help, 301-304

C

Call By Reference Nodes, 274
Call Library Function Nodes, labeling, 305
Capacitor Test & Sort applications, 79
capitalization, applying consistent, 62
centering dialog VI panels, 51
Centrifuge DAQ System VIs, 199-200
Centrifuge DAQ VIs, 122-123, 292
centrifuge data acquisition (DAQ), 81
CHM (Microsoft Compressed HTML Help), 312
Classic State Machine design pattern, 257-258
Clear Mode, 228
clearing errors, 228
Cluster Wired tests, 323
clusters
  errors, 144
  modularization, 97-101
  multiple distinct items, grouping, 308
  shells, 301
  simple arrays and, 181-184
CM (configuration management), 21
CM (content management), 46. *See also* source control
Code and Fix software lifecycle development model, 23
code
  errors, 214, 217
  reuse, 34-39
  reviews, 18, 319
    peer reviews, 334-336
    self reviews, 320-329
  source. *See* source code
Coercion Dots test, 322
colors
  front panels, 69, 71
  icons, selecting, 135
  reusing, 136
Comment Usage errors, 328
comments, applying, 305
commercial subVIs, 48

complex application frameworks, 272
  Dynamic Framework, 272-278
  Modular Multiple-Loop Framework, 283-286
  Multiple-Loop Framework, 278-282
Complexity, 16. *See also* simplicity
components, 90, 272
compound design patterns, 267-272
Compute Statistics VI, 171
conditions, race, 244
configuration management (CM), 21
Configuration tab, 336
configuring
  data constructs, 170, 172
  Data Range property, 72
  data structure properties, 169
  LabVIEW environment, 32
    code reuse, 34-39
    LabVIEW Options dialog box, 32-33
  resolution, 88
  tabbing order, 73
connector panes, 140-148, 150-154
consistency, 74, 165-169
constants, labels, 305
Cont Acq&Graph Voltage-To File(Binary) VI, 230-231
content management. *See* CM (content management)
continuous acquire to file, 230-231
Continuous Loop design patterns, 246-250
Control Alignment tests, 323
Control Editor, 169
controls
  code errors, 214, 217
  data types, 158-172
  descriptions, 301
  grouping, 49
  Interval, 59
  labels, 66-67
  managing, 59
  Mode, 59
  Pause, 59
  scope, 71-73
Controls palettes, 36
converting LLBs to project libraries, 44
copying graphics, 138-139
Curve local variables, 108
custom palettes, 33
customizing run-time menus. *See also* configuring

D

DAQmx Task, 146
Data Communication palette, 223
data constructs, 157, 180-184
  creating, 170-172
  nested, 187-192
  special, 186
data dependency, 106
data flow, 101, 103-114
Data Range property, configuring, 72
data structures, 157
  design methodology, 158, 160-172
  examples of, 193-197
  properties, 169
data types, 157
  simple, 172
    Boolean, 173-174
    numeric, 175-176
    path, 179-180
    picture, 179-180
    special numeric, 177-179
    strings, 179-180
decimal notation, 176
default appearance, 59
default filenames, 44-45
default values, labels, 301
Definition 1.1 (development time), 18
deleting template instructions, 62, 301
Dequeue Element function, 262
descriptions, VIs, 311

design, 29
  data structures, 158, 160-172
  patterns, 239, 241
    Classic State Machine, 257-258
    compound, 267-272
    Continuous Loop, 246-250
    Event Machine, 265-267
    Event-Driven State Machine, 262-264
    Event-Handling Loop, 250-254
    examples, 287-297
    Functional Global, 244-245
    Immediate SubVI, 241-244
    Queued State Machine, 260-262
    state machines, 254-256
  proof of concept, 30-31
  resources for, 30
  revision of specifications, 31
  templates. *See* templates
desktop applications, applying dialogs, 51-52
desktop GUI VIs, 48
development style. *See* style
development time. *See* time
diagrams. *See* block diagrams
Dialog & User Interface palettes, 51
Dialog Controls tests, 323
Dialog Using Events templates, 37
dialog VIs, 47, 51
Dialog Window Appearance, 53
dialogs
  applying, 51-52
  controls, 52
Digital Inputs, 146
disk organization, 40
  file repositories, 40
  folder hierarchies, 40-42
Distribute Objects menu, 50
documentation
  block diagrams, 304-308
  examples of, 315, 318
  front panels, 301-304
  icons, 311
  online, 311-314
  tests, 321
  VI descriptions, 19, 311
double precision (DBL) representation, 175
dynamic data constructs, 187
dynamic event terminals, 266
Dynamic Framework, 272-278
dynamic loaders, 272

E

ease of use, 6-7
efficiency, 7-9, 160-164
Elapsed Time VI, 287
embedding text, 301
enlarging objects, 52-57
Enqueue Element function, 262
entering control descriptions, 301
enums
  applying, 306
  saving, 307
Equation 1.1 (modularity index), 13
Error case, 218
Error Cluster Wired failures, 323-326
errors
  clusters, 144, 218
  error handling, 204
    automatic, 230
    clearing errors, 228
    error codes, 214-217
    examples, 230-238
    merging errors, 226-227
    prioritizing errors, 222-225
    reporting errors, 211-214
    robustness, 12-15
    structure wiring, 226
    subVI, 217-222
    trapping errors, 205-209
  terminals, 109, 218-222, 323-326
evaluation tools for style, 18
Event Machine design patterns, 265-267
Event-Driven State Machine design patterns, 262-264
Event-Handling Loop design patterns, 250-254

events
  efficiency, 8
  registration, 266
  source terminals, 266
  Timeout, 251
  users, 266
  Value Change, 250
Excessively Nested VIs, 117
Execution category, 230
Exit Boolean
  controls, 109
  text, 63
Exit Value Change events, 105-109
Express Input subpalettes, 223

F

Fetch Waveform VI, 167
File Dialog functions, 326
File I/O palette, 36
File Path indicators, 120
files
  naming conventions, 44-45
  repositories, 40
Filled Rectangle tool, 135-136
Flexible Sequencers, 112
folder hierarchies, 40-42
fonts
  applying, 62
  icons, 133
  reusing, 136
  selecting, 63-65
foreground icons, 135
formatting. *See also* design
  block diagrams, 88-93
  data constructs, 170-172
  front panels, 48
    aligning objects, 50
    applying controls, 52
    applying dialogs, 51-52
    avoiding overlapping objects, 57-58
    centering dialog VI panels, 51
    enlarging objects, 52-57
    grouping controls, 49
    importing images, 59
    maximizing top-level VIs, 51
    navigating GUIs, 71-73
    sizing objects, 50-51
    text, 61-71
  icons, 132
frameworks
  applications, 239, 272
    Dynamic Framework, 272-278
    examples, 287-297
    Modular Multiple-Loop Framework, 283-286
    Multiple-Loop Framework, 278-282
front panels
  colors, 69, 71
  documentation, 301-304
  GUI navigation, 71-73
  LabVIEW environment options, 32
  layout, 48
    aligning objects, 50
    applying controls, 52
    applying dialogs, 51-52
    avoiding overlapping objects, 57-58
    centering dialog VI panels, 51
    enlarging objects, 52-57
    grouping controls, 49
    importing images, 59
    maximizing top-level VIs, 51
    sizing objects, 50-51
  maintainability, 11
  readability, 9-11
  tests, 321
  text, 61-69
Function parameters, 177
Functional Global design pattern, 244-245
functions
  Bundle By Name, 308-310
  Dequeue Element, 262
  Enqueue Element, 262
  error handling, 204
    automatic, 230
    clearing errors, 228

error codes, 214, 217
examples, 230-238
merging errors, 226-227
prioritizing errors, 222-225
reporting errors, 211-214
structure wiring, 226
subVIs, 217, 220-222
trapping errors, 205-209
Two Button Dialog, 52
Unbundle By Name, 308-310
Functions palette, 35, 150

G–H

General Error Handler VI, 210-211
Get Raw I16 Waveform VI, 166
global design patterns, 244-245
glyphs
reusing, 136
selecting, 133
graphical user interfaces. *See* GUIs
graphics, copying, 138-139
grouping controls, 49
GUIs (graphical user interfaces), 6, 47
ease of use, 6-7
front panels, navigating, 71-73
guidelines. *See* rules

Hamburger, Bob, 147
Haphazard VIs, 119-120
Help menus, 301-304
hexadecimal notation, 176
hiding
labels, 307
toolbars, 58
HTML (HyperText Markup Language), 312

I–J

I/O (input/output), 146
Datalog palettes, 212
resource efficiency, 7-9
icons, 132-135, 311
creating, 311
examples, 145-154
international, 139-140
shortcuts, 135-139
ID Query controls, 148
Idle state, 256-257
images, importing, 59
Immediate SubVI design patterns, 241-244
importing images, 59
industrial applications, 51-52
industrial GUI VIs, 48, 69
Initialize case, 113, 236
Initialize state, 256
input/output. *See* I/O
instrument drivers
code reuse, 34
file-naming conventions, 45
instrument prefixes, 148
interfaces
connector panes, 140-145
GUIs. *See* GUIs
Parafoil Guidance Interfaces, 83-84
international icons, 139-140
Interval controls, 59
intuitive control labels, applying, 301

K–L

labels
algorithms, 305
comments, 305
constants, 305
controls, 66-67
default values, 301
hiding, 307
loops, 305
subdiagrams, 305
terminals, 304
wiring, 305

LabVIEW environments
  configuring, 32
    code reuse, 34-39
    LabVIEW Options dialog box, 32-33
  GUI Essential Techniques, 48
  .ini file backups, 33
  Options dialog box, 32-33
  project requirements specifications, 25-28
  Tools Network website, 30
LAVA website, 30
layouts
  block diagrams, 88-93
  front panels, 48
    aligning objects, 50
    applying controls, 52
    applying dialogs, 51-52
    avoiding overlapping objects, 57-58
    centering dialog VI panels, 51
    enlarging objects, 52-57
    grouping controls, 49
    importing images, 59
    maximizing top-level VIs, 51
    sizing objects, 50-51
Left Audio Output indicator, 53
Left to Right VI, 121-122
libraries
  code reuse, 35-37
  LLBs
    converting to project libraries, 44
    project libraries versus, 44
  project libraries, 43-44
Load Script case, 236
Log Data case, 120
Log Interval labels, 63
looped Case structures, 111-114
loops
  Continuous Loop design patterns, 246-250
  efficiency, 7-8
  Event-Handling Loop design patterns, 250-254
  labels, 305
  Modular Multiple-Loop Framework, 283-286
  Multiple-Loop Framework, 278-282
  parallel, 269-272
  tight, 279

M

Main tab, 336
maintainability, 11
managing controls, 59
manual checklists, 330-334
Manual Resolution parameters, 177
maximizing top-level VIs, 51
memory
  data types, 160-164
  efficiency, 7-9
Memory Usage VI properties page, 18
menus, Help, 301-304
Merge Errors VI, 232
merging errors, 226-227
message output terminals, 212
Meticulous VIs, 2, 6
  ease of use, 6
  efficiency, 8
  error handling, 13
  maintainability, 11
  modularity index, 13
  performance, 17
  readability, 9
  simplicity, 16
Microsoft Assistance Markup Language (AML), 312
Microsoft Compressed HTML Help (CHM), 312
minimizing text, 62
Mode controls, 59
modifying colors, 69
Modular Multiple-Loop Framework, 283-286
modularization
  clusters, 97, 100-101
  indexes, 13
  subVIs, 90
moving source files, 46
Multiple-Loop Framework, 278-282
multivalued data types, applying, 308

N

naming conventions. *See* file naming conventions
navigation, GUI, 71-74
nested data constructs, 187-192
nested structures, efficiency, 9
Nested VIs, 4
  ease of use, 6
  efficiency, 7-9
  error handling, 14
  maintainability, 12
  modularity indexes, 13
  performance, 17
  readability, 10
  simplicity, 16
New UUT Boolean controls, 336
New UUT command buttons, 337
New UUT events, 336
NI Developer Zone website, 30
NI Example Finder, 30
NI Instrument Driver Guidelines, 148-151
No Error case, 218
nodes, simplicity, 15-17
notification output terminals, 109
numeric data types, 175-176
Numeric palettes, 223

O

objects
  aligning, 50
  default appearance, 59
  enlarging, 52-57
  overlapping, 57-58
  sizing, 50-51
online documentation, 311-314
OpenG
  variants, 194-197
  website, 30
optical filter test application, 127-128
optimizing data flow, 111, 114
optional priority, 144
output. *See* I/O
Output subpalettes, 223
overlapping objects, 57-59
Overview tab, 56

P

palettes
  code reuse, 35-37
  custom palettes, 33
  Dialog & User Interface, 51
  System, 53
panels
  default appearance, 59
  front. *See* front panels
  resizing, 59
  subVI, 48, 68
Parafoil Guidance Interfaces, 83-84
parallel loops, 269-272
Parse and Transmit Dialog VIs, 78
Parse and Transmit VIs, 77
path data types, 179-180
patterns
  design, 239, 241
    Classic State Machines, 257-258
    compound, 267-272
    Continuous Loop, 246-250
    Event Machine, 265-267
    Event-Driven State Machines, 262-264
    Event-Handling Loop, 250-254
    examples, 287-297
    Functional Global, 244-245
    Immediate SubVI, 241-244
    Queued State Machines, 260-262
    state machines, 254-256
  terminals, 140
  VIs, 141
Pause controls, 59
PDF (Portable Document Format), 311
peer reviews, 334-336
performance, 17-18

picture data types, 179-180
Pixel Tolerance parameter, 323
plug-ins, 272
Poll Instrument Response VI, 288
polling, 108
polymorphism, 175
Portable Document Format (PDF), 312
Power Monitor System VI, 54-57
prefixes, instruments, 148
preparation
  design documentation, 29
    proof of concept, 30-31
    resources for, 30
    revision of specifications, 31
  disk organization, 40
    file repository, 40
    folder hierarchy, 40-42
  file-naming conventions, 39-45
  project organization, 39-44
  requirements specifications, 22-24
    best practices, 24-25
    projects, 25-28
  source controls, 39-46
preparations, 32
  code reuse, 34-39
  LabVIEW Options dialog box, 32-33
presentation section (requirements specifications), 28
priority
  errors, 222-225
  inputs/outputs, 144
  requirements specifications, 28
processor resources, efficiency, 7-9
Profile Performance and Memory window, 7, 17-18
Project Explorer, 42
projects
  journals, 24-25
  libraries, 43-44
  organization, 39-43
  requirements specifications, 25-28
proof of concept, 30-31
properties
  data structures, 169
  Visible, 222

Q–R

Queued State Machine design pattern, 113, 260-262
Quit Value Change event, 114, 336

race conditions, 244
radix, visible, 176
rapid code development, 22
Read Waveform VIs, 167-168
readability, 9-11
recommended priority, 145
record type input terminals, 212
references, assigning, 144
reporting errors, 211, 213-214
required priority, 144
requirements specifications, 22-24
  best practices, 24-25
  projects, 25-28
  revision of, 31
  Theorem 2.1, 22
Reset controls, 148
Reset Energy Boolean text, 63
Resize Objects tool, 50
resizing panels, 59
resolution, configuring, 88
resources for design documentation, 30
reuse. *See* code reuse
revision
  numbers, 45
  of requirements specifications, 31
Revision Number labels, 63
Right Audio Output indicators, 53
Right Edges tool, 50
Right to Left VIs, 120
Ritter, David, 48
robustness, 12-15

rules
  applications, providing online documentation, 314
  applying Timed Loops, 246
  arrays
    applying for multivalued data items, 181
    entering descriptions, 182, 301
    storing large or dynamic length data sets, 181-182
  block diagrams
    applying high object density, 89
    configuring resolution, 88
    creating multilayer hierarchy of subVIs, 90
    leaving backgrounds white, 89
    limiting, 89
  Boolean data types
    applying if two states are logical opposites, 173
    assigning names, 173
    configuring parameters, 173
    formatting controls, 174
    labeling, 173
  Call By Reference Node, 274
  clusters
    applying alignment tools, 182
    avoiding for interactive controls, 184
    bundling/unbundling by name, 183
    entering descriptions, 301
    grouping multiple distinct items, 308
    saving, 183
  CM (content management) Rules, 46
  code reviews
    applying enumerated type definitions, 332
    applying symmetry and spacing, 330
    automating inspections, 321
    avoiding clusters, 335
    avoiding polling GUI objects, 332, 335
    avoiding Sequence structures, 332
    avoiding variables, 332
    configuring controls, 330
    configuring Tabbing Order, 335
    consistent fonts and capitalization, 330
    customizing test criteria, 321
    designating neutral parties to take notes, 335
    enlarging objects, 330
    General Error Handler VI over Simple Error Handler VIs, 332
    grouping controls, 330
    implementing error log files, 335
    including states for Initialize, Idle, Shutdown and Blank, 332
    labeling wires, 332
    limiting controls, 330
    limiting information on GUIs, 330
    minimizing wire bends, eliminating kinks and loops, 332
    modifying source code, 335
    peer reviews, 320, 334
    prioritizing rules, 323
    selecting memory efficient data types, 335
    style conventions, 320
    suppressing dialog error reporting, 335
    tunneling wires, 332
  colors, 69
    creating themes, 70
    following universal conventions, 70
    leaving panels/objects subVI gray, 70
    modifying, 69
    simplifying color schemes, 71
  comments
    applying, 305
    leaving notes for development teams, 306
  components
    maintaining, 274
    passing data between shift registers, 275
  connector panes
    applying 4×2×2×4 patterns, 142
    assigning controls, 142
    assigning error clusters, 144
    assigning references, 144
    avoiding crossing wire stubs, 142
    selecting left and right vertical edge connector terminals, 144
    selecting patterns with extra terminals, 140
    selecting top and bottom horizontal edge connector terminals, 144

selecting unified patterns for VIs, 141
specifying priority for inputs/outputs, 144
specifying terminal assignments, 144
controls, 52
applying, enum, ring, and path controls, 180
applying ring or enumerated, 72
arranging, 59
avoiding strings, 179
combining labels, 68
configuring default values, 169
configuring tabbing order, 73
Data Range property, 72
entering descriptions, 169, 301
grouping, 49
including Help buttons/menus, 73
labels, 66-67
limiting, 71
passing references, 283
placing within Value Change event cases, 251
providing default vales, 68
restricting ranges, 71
saving custom, 169
data constructs, creating arrays and clusters, 170, 172
data flow
always flow left to right, 102
applying Flat Sequence structures, 105
applying global variables for simple data sharing, 105
applying looped Case structures, 111-114
applying shift registers, 111
avoiding array and cluster coercions, 103
avoiding nesting beyond 3 layers, 104
avoiding sequence structures, 103-104
avoiding variables, 108
creating controls and constants, 103
disabling dots at wire junctions, 103
grouping shift registers, 111
initializing control values, 105
labeling wires exiting left shift register terminals, 111
ordering operations, 105
propagating error clusters, 102
data structures
selecting consistent data types, 165-169
selecting controls, 158
selecting memory efficient data types, 160-161, 164
data types, applying multivalued, 308
dialog VI panels
centering, 51
sizing, 51
enums, applying, 306
error handling
applying Error Case Structure, 218-220
applying Error Handling templates, 222
applying negative codes of I/O device errors, 217
applying unwired defaults over constants, 220-222
avoiding subVIs with built-in error reporting, 213
disabling indexing, 208-209
General Error Handler VI, 210-211
implementing error logs, 211-213
maintaining user-defined error codes, 217
propagating error clusters, 205
reporting, 205, 210
suppressing dialog error reporting, 213-214
trapping errors, 224-226, 230
trapping from all iterations of loops, 206
trapping from nodes, 209
events
avoiding continuous timeouts, 251
resizing data nodes, 251
Value Change, 250
files
naming conventions, 44-45
repositories, 40
folder hierarchies, 40-42
fonts
applying 13 point black, 68
applying consistent, 62
selecting, 63, 65
functions, Bundle and UnBundle By Name, 308-310

GUIs
  limiting information, 54
  navigation, 71-73
Help menus, 301-304
icons
  applying black borders, 132
  avoiding default icons, 132
  avoiding text and graphics, 139
  budgeting time for intended VI reuse, 135
  combining glyphs with colors and text, 132
  copying graphics, 138-139
  creating, 132, 311
  creating templates for, 136
  formatting fonts, 133
  reusing glyphs, 136
  saving VIs, 132
  selecting colors, 135
  selecting glyphs, 133
  selecting unified styles, 134
images, importing, 59
industrial GUI VIs
  large text size for command buttons, 69
  maximizing text colors, 69
labels
  algorithms, 305
  constants, 305
  hiding, 307
  loops, 305
  providing default values, 301
  subdiagrams, 305
  viewing terminal, 304
  wiring, 305
LabVIEW.ini backups, 33
layers, avoiding nesting beyond 3, 241
loops
  applying multiple criteria, 246
  applying shift registers, 246
  avoiding subVI selection for continuous, 283
  creating parallel, 278
  grouping shift registers, 246
  including, 279
  including delays, 246
  labeling, 270
  maintaining Event-Handling at top-level, 284
  messaging between, 269
  minimizing space, 270
  prioritizing, 269
  sizing, 270
modularize high-level subVIs into lower level subVIs, 90
nested data structures
  avoiding manipulating, 188
  limiting size of arrays, 188-192
  organizing complex data, 188
numeric data types
  applying automatic formatting, 176
  applying I32 representations, 175
  formatting, 176
objects
  aligning, 50
  avoiding overlapping, 57
  avoiding polling GUI, 250
  default appearance, 59
  enlarging and centering, 52
  sizing, 50
panels
  resizing, 59
  text, 61-69
projects
  journals, 24-25
  libraries, 43-44
proof of concept, style and, 31
reusable libraries, 35-37
reusable subVIs, 35
reusable templates, 38-39
run-time menus, customizing, 73
sequences, avoiding structures, 241
source file movement, 46
special numeric data types
  applying enums and rings, 178
  saving enums and rings, 179
state machines
  applying, 255
  applying enumerated type definitions, 255
  avoiding timeout, 262
  deriving application primary states, 255
  dividing primary states, 255

    including Initialize, Idle, Shutdown, and Blank, 256
    minimizing external code, 256
  subVIs
    avoiding trivial containing few nodes, 91
    creating meaningful icons, 93
    creating to save space, 91
    modularizing top-level diagrams, 241
  templates, deleting instructions, 62, 301
  terminals
    assigning, 275
    selecting connectors, 274
  text
    applying default 13-point plain application fonts, 305-306
    embedding, 301
    minimizing, 62
    minimizing front panels, 301
  toolbars, hiding, 58
  top-level VIs, 45, 51
  updates, maintaining high speed display at top-level, 284
  wiring
    avoiding local and global variables, 96
    avoiding structures, 96
    labeling, 96, 246
    limiting length, 96
    locating unwired front panel terminals, 97
    maintaining spacing of parallel wires, 94
    minimizing bends, 94
    modularizing into clusters, 97, 100
    obstructing view of, 96
    saving clusters as type definitions, 101
    tunneling, 96
  written requirements specifications, 25
Run Test Boolean controls, 105
Run Test events, 109, 113, 267
run-time menus, customizing, 73

S

saving enums, 307
schemes, colors, 71
scope
  controls, 71-73
  creep, 22
  Theorem 2.1, 23
Screw Inspection VIs, 124, 126, 234-235
Select Tests dialog, 322
selecting
  controls, 158, 160-172
  fonts, 63-65
  icons, 133
  subVIs, 75-76
Selection utility, subVIs from, 115-117
self reviews, 320
  manual checklists, 330-334
  VI Analyzer Toolkit, 320-323, 329
Sensor Scaling clusters, 105
Sensor Scaling write and read local variables, 108
Sensor Selection listbox controls, 199
Sequence structures, 104-109
Serial Port Settings cluster, 148-149
Serial Ports, 148
shift registers, efficiency, 8
shortcuts icons, 135-136, 139
Shutdown state, 114, 256, 267
simple arrays, 181-184
simple controls, 158
simple data structures, 158
simple data types, 172
  Boolean, 173-174
  numeric, 175-176
  path, 179-180
  picture, 179-180
  special numeric, 177-179
  strings, 179-180
Simple Error Handler VIs, 210-211
simplicity, 15-17
single-precision (SGL), 175
sizing objects, 50-51
SMTP Email palette, 213
source code
  maintainability, 11
  peer reviews, 334-336
  readability, 9-11
  self reviews, 320

manual checklists, 330-334
VI Analyzer Toolkit, 320-323, 329
source control, 39-46. *See also* CM (content management)
source files, moving, 46
Spaghetti VIs, 5
ease of use, 6
efficiency, 7
error handling, 14
maintainability, 12
modularity index, 13
performance, 17
readability, 11
scope creep, 23
simplicity, 16
special data constructs, 186
special numeric data types, 177-179
specifications. *See* requirements specifications
specifying terminal assignments, 144
Spectralyzer, 82-83
Standard State Machine design patterns, 37
standard subVIs, 48
state machines, 254-256
Classic State Machine design patterns, 257-258
Event Machine design patterns, 265-267
Event-Driven State Machine design patterns, 262-264
Queued State Machine design patterns, 260-262
Statistics clusters, 324
Stop global variables, 105
strict type definitions, 101
string data types, 179-180
structures
data. *See* data structures
Sequence, 104-109
wiring, 226
style
ease of use, 6-7
efficiency, 7-9
evaluation tools, 18
icons, 132-135
importance of, 2-6
maintainability, 11
performance, 17-18
proof of concept and, 31
readability, 9-11
robustness, 12-15
simplicity, 15-17
Theorem 1.0, 2
Theorem 1.1, 19
time versus, 18-19
subdiagrams, labels, 305
subVIs
code reuse, 35
efficiency, 9
error handling, 37, 217, 220-222
from Selection utility, 115-117
icons, 311
modularization, 90
panels, 48, 68
performance, 17
robustness, 12-15
selecting, 75-76
utility VIs, 36
Suss Interface Toolkit, 137, 231-232
symbolic mappings, 63
Synchronization palettes, 223
System palettes, 53

T

tabbing order, configuring, 73
tabs, overview, 56
task ID out 2, 146-147
templates
code reuse, 36-39
Error handling, 222
icons, 136
instructions, 62, 301
terminal labels, viewing, 304
terminals
assigning, 144-154
interfaces, 140-145
top and bottom horizontal edge connectors, 144
vertical edge connectors, 144

Test Executive VIs, error handling, 235-238
Test to Run numeric control, 306
tests
  documentation, 321
  Front Panels, 321
  methodology section (requirements specifications), 28
text
  default
    13-point plain black, 305-306
    appearance, 59
  embedding, 301
  front panels, 61-69, 301
  industrial GUI VIs, 69
  minimizing, 62
  subVIs, 68
themes, creating, 70
Theorem 1.1 (style elements), 2
Theorem 1.2 (style and development time), 19
Theorem 2.1 (requirements specifications), 22
Theorem 2.2 (scope creep), 23
Thermometer VI, 193-194
tight loops, 279
time
  style versus, 18-19
  Theorem 1.2, 19
Timeout events, 251
Timing palettes, 212, 223
toolbars, hiding, 58
tools
  Bottom Edges alignment, 95
  Filled Rectangle, 135-136
  Resize Objects, 50
  Right Edges, 50
  Vertical Compress, 50
top and bottom horizontal edge connector terminals, 144
Top level application window, 58
top-level VIs, 47
  arrays, bundling, 167
  identifying, 45
  maximizing, 51
Torque Hysteresis VI, 171-172, 189-191, 214, 330, 336
transducers, 293, 295
trapping errors, 205-209
Two Button Dialog function, 52
type definitions, 101
type of dialog input terminals, 212

U

Unbundle By Name function, 183, 308-310
unified patterns, selecting for related VIs, 141
unit under test (UUT), 170
Unused Code failures, 323, 326
Use transparent name labels, 67
User Controls palette, 36
user event data type input terminals, 266
user events, 266
User Interface category, 323
User Libraries palette, 36
utility VIs, 36
UUT (unit under test), 170

V

Value Change events, 250
variables, data flow, 104-109
variants, 187, 194-197
velocity, 324, 328
Vertical Compress tool, 50
vertical edge connector terminals, 144
VI Analyzer Toolkit, 320-323, 329
VI descriptions, style versus time, 19
VI Documentation failures, 327
VI Hierarchy windows, 166
VI History, style versus time, 19
VI Metrics window, 12, 15, 18
VI Properties dialog, 230
VI Properties window, 8
View as Icon (shortcut menu), 152
VIs
  descriptions, 311
  error handling, 204

automatic, 230
clearing errors, 228
error codes, 214, 217
examples, 230-238
merging errors, 226-227
prioritizing errors, 222-225
reporting errors, 211-214
structure wiring, 226
trapping errors, 205-209
templates. *See* templates
top-level VIs
arrays, bundling, 167
identifying, 45
maximizing, 51
Visible property, 222
visible radix, 176

W–Z

Wait n mSec VIs, 225
WDT (waveform data type), 186
Window Appearance property pages, 58
Wire Bends tests, 324, 329
Wires Under Objects errors, 323, 327
wiring, 93-101
connector pane, 140-145
labels, 305
structures, 226
written specifications. *See* requirements specifications