

Stichwortverzeichnis

Cross-Referenz der Heuristiken

C1 320, 323, 337
C2 323, 327, 336, 338
C3 326, 327, 330, 338
C4 338
C5 339
E1 339
E2 339
F1 278, 340
F2 340
F3 340
F4 318, 329, 340
G1 320, 340
G10 326, 345
G11 309, 326, 329, 332, 345
G12 327, 329, 330, 336, 346
G13 328, 329, 346
G14 329, 330, 333, 346
G15 330, 347
G16 330, 331, 348
G17 330, 349
G18 331, 332, 350
G19 331, 332, 333, 350
G2 318, 319, 341
G20 332, 351
G21 332, 351
G22 334, 352
G23 278, 335, 353
G24 336, 354
G25 336, 354
G26 355
G27 356
G28 307, 356
G29 308, 356
G3 319, 341
G30 309, 356
G31 310, 357
G32 310, 358
G33 312, 359
G34 359
G35 361

G36 362
G4 322, 341
G5 322, 328, 332, 336, 342
G6 323, 326, 330, 334, 335, 336, 342
G7 324, 344
G8 326, 344
G9 313, 326, 327, 329, 345
J1 320, 362
J2 321, 363
J3 326, 327, 364
N1 310, 321, 323, 325, 329, 330,
331, 335, 336, 366
N2 321, 367
N3 327, 330, 368
N4 308, 332, 368
N5 258, 369
N6 307, 369
N7 308, 370
T1 319, 370
T1, 16 318
T2 318, 370
T3 318, 370
T4 371
T5 319, 371
T6 319, 371
T7 319, 371
T8 320, 371
T9 371

A

Abhängigkeit
logische vs. physische 352
Abhängigkeitsmagnet 79
Error.java 79
Abschnitt
kritischer 221
Abstand
vertikaler 114
Abstract Factory 54, 69
Abstract Factory-Pattern 194, 324
Abstraktion 130

- Abstraktionsebene 26, 66, 192, 342, 359
 - Funktion 67
 - Namen 367
- Abteilung 192
- Acceptance-Test 256
- Active Record 137
- Adapter 226, 386
- Ad-hoc-Code 159
- Affinität
 - konzeptionelle 118
- Agile Conference 465
- Agilität 196
 - Praktiken 208
- Algorithmus 351
- Änderung
 - einplanen 185
 - isolieren 188
- Annotation 205
- Anordnung
 - vertikale 119
- Anti-Symmetrie 131
- Anwendungsinfrastruktur 202
- AOP siehe Aspect-oriented Programming
- Architektur 15
 - inkrementelle 196
 - naiv einfache 206
 - Software-Entwicklung 16
 - System 196
- Args 235
- Argument
 - Anzahl 71
 - Flag-Argumente 72, 340
 - hängendes 347
 - in Funktionen 340
 - Input-Argumente 71, 72
 - Listen 74
 - Objekte 74
 - Output-Argumente 71, 72, 76, 340
 - Reihenfolge 73
 - Selektor-Argumente 347
 - Whitespace 120
- ASM 200, 231
- AspectJ 205
- Aspect-Oriented Framework 231
- Aspect-oriented Programming 199
- Aspekt 199, 205
- Aspektorientierte Programmierung siehe Aspect-oriented Programming
- assertEquals 73
- AtomicBoolean 384
- AtomicInteger 384
- AtomicReference 384
- Aufgabe
 - eine 65
 - mehr als eine 193
- Aufruf-Stack 381
- Aufwärtsskalierung 196
- Ausdrucksstärke 214
- Ausfallsicherheit 202
- Ausnahme 139
 - checked 143
 - Klassifikation 144
 - mit Kontext auslösen 144
 - oder Fehler-Code? 77
 - unchecked 143
- Ausrichtung
 - horizontale 121
- Ausschluss
 - gegenseitiger 395
- Autor
 - Programmierer 41
- B**
- Basisklasse 344
- BDUF siehe Big Design Up Front
- Bean 136, 197, 203
- Beck, Kent
 - 27, 37, 64, 105, 210, 301, 342, 350, 465
- Bedingung
 - Kapselung 356
 - negative 356
- Befehlszeilenargument 235
- Belang siehe Concern
- Belang siehe Responsibility
- Benutzer-Story 206
- Big Design Up Front 206
- Booch, Grady 34
- Boundary-Interface 153
- Boundary-Test 157
- Bowling game 369
- Brooks, Fred 17
- Bucket Brigade 358
- Build 339
- Build-Operate-Check(Pattern) 165
- Byte-Code-Manipulation 200

C

Caching 202
 Call stack 381
 Callable 383
 CGLIB 200, 231
 COBOL 354
 Codd-Normalform 342
 Code 26

- Ad-hoc-Code 159
- als Erklärungsmedium 87
- als Spezifikation 26
- Ausdrucksstärke 214
- auskommentierter 102, 339
- Drittanbieter-Code 151
- duplizierter 61, 211
- Duplizierung 38
- Gründe für Verrotten 30
- Instrumentierung 230
- Lesbarkeit 34, 42
- literate 36
- Merkmale von sauberem 32
- minimal 35
- noch nicht existierender 157
- prozeduraler 133
- Redesign 29
- Rohentwurf 249
- sauberer 22
- schlechter 27
- schmutziger 243
- Shutdown-Code 227
- shy Code 362
- Threaded-Code 229
- toter 345
- und Design 19
- und Sprache 26
- vor Konsequenzen warnen 90
- Zusammenhang mit Kommentar 104
- Zweck erklären 88

 Code Smells 337
 Code-Coverage 336
 Code-Gefühl 31
 Codieren 22
 Collection

- thread-sichere 223

 Concern 192

- Cross-Cutting 199, 202
- Fehler-Handling 150
- Trennung 192

 Constantine, Larry 19
 ConTest 232, 401
 Convention over Configuration 203

Coplien, James O. 217
 Coverage-Analyse 304
 Coverage-Tool 370
 Cross-Cutting Concern 199, 202
 Cruft 17
 Cunningham, Ward 38, 70

D

DAO siehe Data Access Object
 Data Access Object 203
 Data Transfer Object 136, 199
 Daten

- Anti-Symmetrie 131
- gemeinsame Nutzung 222
- relationale 197

 Datenabstraktion 129
 Datenstruktur

- Unterschied zu Objekt 131

 Datentransfer-Objekt 136
 Datentransferobjekt siehe Data Transfer Object
 Datenzugriffs-Objekt 203
 DBMS 199
 Deadlock 224, 393, 396
 Decorator 368
 Denkschule 40
 Dependency Injection 195, 202, 210
 Dependency-Inversion-Prinzip 190
 Dependency-Management 195
 Deployment-Deskriptor 198
 Design 210

- und Code 19

 Design Pattern 38
 Details

- in der Software-Entwicklung 17

 DI siehe Dependency Injection
 Dichotomie

- fundamentale 133

 Dichte

- vertikale 113

 Dijkstra, Edsger 80
 Dining-Philosophers-Problem 225
 DIP 209
 DIP siehe Dependency-Inversion-Prinzip
 Domain-Specific Language 208
 Don't-Repeat-Yourself-Prinzip 80
 Drittanbieter-Code 151
 DRY siehe Don't-Repeat-Yourself-Prinzip
 DRY-Prinzip 342, 388
 DSL siehe Domain-Specific Language
 DTO 136

DTO siehe Data Transfer Object
 Dummy-Bereich 124
 Duplizierung 38, 211, 342
 Durchsatz
 berechnen 392
 Dyade 71

E

Eclipse 35, 37, 56
 Eimerkette 358
 Einhüllen 145
 Einkapselung 174
 Einmal, und nur einmal 342
 Einrückung 123
 durchbrechen 124
 EJB1 197
 EJB2 197, 198, 203, 205
 EJB3 203
 EJB-Container 218
 Emergenz 209
 Entity Bean 197
 Entscheidung
 aufschieben 207
 Entwicklung
 inkrementelle 256
 schrittweise 258
 Error.java
 Abhängigkeitsmagnet 79
 Erweiterbarkeit 110
 Erzeuger-Verbraucher-Problem 224
 Evans, Eric 368
 Event 72
 Exception siehe Ausnahme
 Exception-Klasse 144
 Executor Framework 383
 Explaining Temporary Variables 331
 Extreme Programming 37, 38, 342

F

F.I.R.S.T. 171
 Factory 194
 Faulheit 356
 Feathers, Michael 36, 139
 Feature Envy 135
 Fehler-Code 79
 oder Ausnahme? 77
 Fehler-Handling 32, 139
 Concern 150
 null zurückgeben 147
 Trennung von Geschäftslogik 146
 Fehler-Verarbeitung 79

FIT 38, 359
 FitNesse
 61, 63, 256, 270, 350, 358, 359, 361, 368
 Kommentar 88, 92
 FitNesseExpediter
 lange Variablenliste 122
 Flag-Argument 72, 340
 Formatierung 109
 Einrückung 123
 horizontale 119
 horizontale Ausrichtung 121
 Regeln 125
 Regeln im Team 125
 vertikale 110
 vertikale Anordnung 119
 vertikale Dichte 113
 vertikale Offenheit 112
 vertikaler Abstand 114
 Zeitungs-Metapher 111
 Zweck 109
 FORTRAN 354
 Fowler, Martin 337, 346
 Frame 381
 Funktion 61, 340
 abhängige 116
 Abschnitte 66
 Abstraktionsebene 67, 359
 Anweisung und Abfrage trennen 77
 Anzahl der Argumente 340
 Anzahl der Aufgaben 356
 Argumente 71
 Argument-Listen 74
 Aufgabenumfang 65
 Blöcke und Einrückungen 65
 drei Argumente 73
 dyadische 73
 ein Argument 72
 Flag-Argumente 72
 Größe 64
 Namen 70
 Objekte als Argument 74
 Output-Argumente 71, 76
 private 345
 tote 340
 zwei Argumente 73
 Funktions-Header
 Kommentar 104
 Funktionsname 351
 Funktionsneid 135, 346
 Future 383

G

Gabriel, Richard 20
 Gamma, Eric 301
 Gefühl für den Code 31
 Geltungsbereich
 Hierarchie 123
 vertikaler 345
 Geschäftslogik 198
 Trennung von Fehler-Handling 146
 Gesetz von Demeter 133, 362
 Getter 129
 Gilbert, David 317
 given-when-then-Konvention 169
 Global 75
 goto 80
 Gott-Klasse 174
 Grenning, James 151
 Grenzbedingung 341, 359
 testen 371
 Grenze 151
 Griffige Abstraktion 34

H

Hierarchie
 Geltungsbereiche 123
 HN siehe Notation, ungarische
 HTML 360
 HTML-Kommentar 102
 Hungarian Notation siehe Notation, ungarische
 Hunt, Andy 33, 342
 Hybrid 137

I

IBM 401
 Implementation Patterns 27
 Implementierung
 verbergen 130
 Implizität 46
 Importliste 362
 Injektion der Abhängigkeit siehe Dependency Injection
 Inkonsistenz 345
 Instanzvariable 115, 354
 Instrumentierung
 von Code 230
 IntelliJ 56
 Interface
 Boundary-Interface 153
 breites 344
 Intuition 341

Inversion of Control 195
 IoC siehe Inversion of Control
 Iterativ und inkrementell 196

J

jar-Datei 344
 Java-AOP-Framework 202
 Javadoc 92
 Kommentar 105
 Java-Proxy 200
 Javassist 200
 JBoss 202
 JBoss AOP 202, 205
 JCommon 317
 JDBC 203
 Jeffries, Ron 37, 342
 Jiggling 232
 JNDI 195
 JUnit 114, 256
 JUnit-Framework 301
 Just-in-Time-Compiler 220
 Just-in-Time-Entscheidung 207
 JVM 197

K

Kapselung
 Bedingung 356
 Kartesische Koordinaten 73
 Kernighan, Brian 85
 Klasse 173
 abgeleitete 344
 Abhängigkeit 344
 Basisklasse 344
 Größe 174
 Namen 55
 nicht thread-sichere 385
 Klassenaufbau 173
 Klassenname 354
 Knuth, Donald 179
 Kohäsion 73, 178, 210
 Kommentar 85, 337
 auskommentierter Code 102
 FitNesse 88, 92
 Funktions-Header 104
 Geschwätz 97, 99
 Grund für Kommentarer 86
 guter 87
 hinter schließenden Klammern 101
 HTML-Kommentare 102
 informierender 88
 irreführender 95

- Javadoc 105
- juristischer 87
- Nebenbemerkung 101
- nicht-lokale Informationen 103
- Positionsbezeichner 100
- redundanter 93, 338
- schlecht geschriebener 338
- schlechter 92
- Tagebuch-Kommentare 96
- TODO-Kommentare 91
- überholter 338
- und schlechter Code 86
- ungeeigneter 337
- ungenauer 86
- vorgeschriebener 96
- Zusammenhang mit Code 104
- Zweck des Codes 88
- Konfigurationswert 361
- Können 21
- Könnerschaft 21
- Konstante
 - Vererbung 363
 - Verwaltung auf Stufen 118
 - vs. Enums 364
- Kontext
 - globaler 193
- Konvention 356
 - Namen 354, 368
- Kopplung 210, 344
 - künstliche 346
 - zeitliche 75, 76, 310, 357

L

- Law of Demeter 133, 362
- Lazy Initialization 192
- Lea, Doug 223, 402
- Lean
 - Prinzipien 16
- Learning Tests 154
- Least-Surprise-Prinzip 349
- LeBlanc's Gesetz 28
- Lern-Test 154, 156
- Lesbarkeit 34, 42, 110
- Leser-Schreiber-Problem 225
- Literate Programming 36, 179
- Livelock 224, 397
- LocalHome 198
- Locking
 - clientbasiertes 226
 - serverbasiertes 226
- LoD 133, 362

- log4j 154
- Logger 154

M

- Map 152
 - clear 152
- Mehrdeutigkeit 356
 - Test 371
- Mentales Mapping 54
- Methode
 - falsche Deklaration 350
 - Namen 55
- Methoden
 - synchronisierte 226
- Methodenname 354
- Mies van der Rohe, Ludwig 15
- Minimaler Code 35
- Missverständnis 356
- Mock Object 193
- Modul 344
- Monade 71, 72
- Monte-Carlo-Test 400
- MQ-Verbindung 394
- Müll 346
- mutual exclusion 395
- Mutual-Exclusion 224

N

- Name 366
 - aussprechbarer 50
 - Auswahl 366
 - Codierungen 52
 - eindeutiger 368
 - Fehlinformationen 47
 - Funktion 70, 351
 - Humor vermeiden 55
 - Implementierungen 54
 - Interfaces 54
 - Klasse 354
 - Klassen 55
 - Kontext 57, 60
 - Konvention 354
 - Länge 369
 - Leerwörter 49
 - Lösungsdomäne 57
 - Member-Präfixe 53
 - Methode 354
 - Methoden 55
 - mit Zahlenserien 49
 - Nebeneffekte 370

- Problemdomäne 57
 - Schlüsselwort 75
 - suchbarer 51
 - Unterschiede verdeutlichen 49
 - Variable 354
 - Wortspiele 56
 - zweckbeschreibender 45
- Name-Mangling 370
- Navigation
 - transitive 362
- Nebeneffekt 75
- Nebenläufigkeit 218, 373
 - EJB-Container 218
 - Mythen 219
 - Probleme 220
- Newkirk, Jim 154
- Nilade 71
- Nomenklatur 368
- Normalform 342
- Notation
 - ungarische 52, 348, 369
- Null 147, 148
- Nutzung
 - gemeinsame Datennutzung 222
- O**
- Objekt
 - Anti-Symmetrie 131
 - Unterschied zu Datenstruktur 131
- Objekt Orientierung 38
- OCP siehe Open-Closed-Prinzip
- Offenheit
 - vertikale 112
- One-off 233
- One-Switch-Regel 353
- OO-Design 177, 188
- Open-Closed-Prinzip 69, 188
 - Checked Exceptions 143
- Operanden-Stack 381
- Operation
 - atomare 380
- OTI 35
- Ottinger, Tim 45
- Output-Argument 71, 76, 340
- Ozzie, Ray 191
- P**
- Persistenz 202
- Persistenzstrategie 199
- Pfadfinder-Regel 43, 307
- Philosophenproblem 225
- PI 349
- PL/I 354
- Platzhalter 362
- Plauger, P. J. 85
- POJO 201, 202, 205
- Polyade 71
- Polymorphismus 353
- Pragmatic Programmer 362
- Prinzip der geringsten Überraschung 349
- Producer-Consumer-Problem 224
- Produktionsumgebung
 - vs. Test-Umgebung 168
- Produktivität 28
- Professionalität 300
- Programmierer
 - als Autor 41
- Programmierung
 - Begriff 26
 - strukturierte 80
- Python 242
- Q**
- Quelldatei
 - Sprache 340
- Quick und dirty 161
- R**
- Readers-Writers-Problem 225
- Redesign 29
- Refactoring 196, 211, 256, 281
- Reihenfolge
 - Argumente 73
- Responsibility 174, 176
- Rohentwurf 243
- Ruby 242
- Rückgabe-Code 139
- Runnable 383
- Run-on-Ausdruck 348
- S**
- Sauberer Code 22
 - Wie schreiben? 31
- Scheren-Regel 115
- Schlechter Code 27
 - Kosten 28
- Schlüsselwort 75
- Schuchert, Brett L. 373
- Scissors-Regel 115
- Scrum 20

- Seiketsu 17
 - Seiri 16
 - Seiso 17
 - Seiton 16
 - Selektor-Argument 347
 - SerialDate
 - Autor 317
 - Refactoring 318
 - Server-Anwendung 373
 - Servlet 222
 - Setter 129, 195
 - Setup-Idiom 193
 - Shutdown-Code 227
 - Shutsuke 17
 - Shy Code 362
 - Sicherheit 202
 - Sicherung
 - übergangene 341
 - Simple Design
 - Regeln 210
 - Single-Responsibility-Prinzip
 - 68, 176, 186, 188, 193, 209, 221, 300, 377
 - Smalltalk 38, 242
 - Smells 337
 - Software physics 206
 - Software-Entwicklung
 - Architektur 16
 - Details 17
 - Softwarephysik 206
 - Software-Projekt
 - Hauptkosten 214
 - Sonderfall 341
 - Sparkle 64
 - Special Case (Objekt) 147
 - Special Case (Pattern) 147
 - 5S-Philosophie 16
 - Sprache
 - domänenspezifische 81
 - in Quelldatei 340
 - Spring 195, 202, 203
 - Spring AOP 202, 205
 - SRP siehe Single-Responsibility-Prinzip
 - Stack-Trace 144
 - Standard 207
 - Startup-Prozess 192
 - Starvation 224, 397
 - Stepdown-Regel 67
 - Strategy (Pattern) 342
 - Stroustrup, Bjarne 32
 - Struktur 356
 - hybride 135
 - Strukturierte Programmierung 80
 - Switch-Anweisung 68
 - Synchronisierte Methoden 226
 - synchronized 221
 - System 191
 - Anwendung 192
 - Konstruktion 192
 - Systemarchitektur 196
 - testen 205
- T**
- Tabellenzeile 197
 - TANSTAAFL 398
 - TDD siehe Test Driven Development
 - Team
 - Produktivität 28
 - Team-Regel 125
 - Template Method (Pattern) 169, 342
 - Template Method-Pattern 213
 - Test 370
 - Anzahl der assert-Anweisungen 168
 - Anzahl der Konzepte 170
 - assert-Anweisung 169
 - Doppelstandard 166
 - Einfachheit 163
 - F.I.R.S.T. 171
 - Geschwindigkeit 171
 - Grenzbedingung 371
 - Klarheit 163
 - Lesbarkeit 163
 - sauber halten 161
 - sauberer 163
 - selbst-validierender 171
 - trivialer 370
 - Unabhängigkeit 171
 - unzureichender 370
 - Wiederholbarkeit 171
 - zeitgerechter 171
 - Test Driven Development
 - 35, 142, 159, 196, 256
 - drei Gesetze 160
 - Testcode
 - Bedeutung 162
 - Testfall
 - als Dokumentation 302
 - Testgesteuerte Entwicklung siehe Test Driven Development
 - TestNG 115, 370

Testsprache 166
 domänenspezifische 166
 Test-Umgebung
 vs. Produktionsumgebung 168
 Thomas, Dave 342
 Thomas, Dave (Big) 33, 35
 Thread
 Unabhängigkeit 222
 Threaded-Code 229
 Threading 218
 Durchsatz berechnen 392
 Thread-Management 377
 Tiger-Team 29
 TO-Absatz 66
 Tomcat 94
 Total Productive Maintenance 16
 TPM 16
 Train Wreck 134
 Transaktion 202
 Trennung
 Concerns 192, 196, 199, 206
 Geschäftslogik und Fehler-Handling
 146
 Verantwortlichkeiten 192
 vertikale 345
 Zuständigkeiten 192, 299
 Triade 71, 73
 Try/Catch-Block 78
 try-catch-finally-Anweisung 141

U

Ubiquitous language 368
 Umkehrung der Kontrolle siehe Inversion of
 Control
 UM-ZU-Absatz 66, 67
 Uncle Bob 40, 379
 Ungenauigkeit 356
 Unit-Test 141, 159, 256, 339
 automatisierter 160
 Unordnung 346

V

Variable
 Aussagekraft 350
 Instanzvariable 115
 Kontrollvariable für Schleifen 115
 lokale 114, 345, 381
 protected 114
 Variablendeklaration 114
 Variablenname 354
 Verantwortlichkeit 176
 Verantwortlichkeit siehe Concern
 Verb 75
 Verfeinerung
 schrittweise 235
 Verhalten
 offensichtliches 341
 Verhungern 397
 Vorausplanung 196

W

Wading-through-Code 28
 Wartbarkeit 110
 Whitespace
 Argumente 120
 horizontaler 120
 Wiki 38, 256
 Wirfs-Brock, Rebecca 464
 Wort
 leeres 49
 Wrapper 145

X

XHTML 360
 XP Immersion 105

Z

Zahl
 magische 348, 354
 Zeitungs-Metapher 111
 Zerlegung 299
 Zugkatastrophe 134
 Zusicherung 149
 Zuständigkeit
 falsche 349
 Zuständigkeit siehe Concern
 Zuständigkeit siehe Responsibility