

Inhalt

| | | |
|--|---|-----------|
| Teil I Grundlagen der Softwarewartung | | 1 |
| 1 | Einleitung | 3 |
| 1.1 | Demografie in der Software..... | 3 |
| 1.2 | Größe als Herausforderung..... | 6 |
| 1.3 | Ursachen für Altersschwäche..... | 11 |
| 1.4 | Wegweiser durch dieses Buch..... | 12 |
| 2 | Begriffswelt der Softwarewartung | 17 |
| 2.1 | Warum Wartung nötig ist..... | 17 |
| 2.1.1 | Entropie..... | 18 |
| 2.1.2 | Lack of movement..... | 19 |
| 2.1.3 | Ignorant surgery..... | 20 |
| 2.2 | Begriffe rund um die Wartung..... | 21 |
| 2.2.1 | Software..... | 21 |
| 2.2.2 | Softwarewartung..... | 23 |
| 2.2.3 | Wartbarkeit..... | 25 |
| 2.2.4 | Softwareevolution..... | 26 |
| 2.3 | Kategorien der Wartung..... | 28 |
| 2.3.1 | Korrektive Wartung..... | 29 |
| 2.3.2 | Präventive Wartung..... | 30 |
| 2.3.3 | Adaptive Wartung..... | 30 |
| 2.3.4 | Perfektionierende Wartung..... | 31 |
| 2.3.5 | Reaktive versus proaktive Wartung..... | 31 |
| 2.3.6 | Support..... | 32 |
| 2.3.7 | Verteilung der Wartungsaufwände..... | 33 |
| 2.4 | Zusammenfassung..... | 34 |

| | | |
|---|--|-----------|
| 3 | Der Lebenszyklus – Grundlage der Wartung | 37 |
| 3.1 | Vom Projekt zum Produkt | 37 |
| 3.2 | Wartung beginnt in der Entwicklung | 39 |
| 3.3 | Ordnung durch Rollen und Organisation | 41 |
| 3.4 | Erfolg durch Prozesse | 42 |
| 3.5 | Zusammenfassung | 44 |
| 4 | Evolution von Software | 47 |
| 4.1 | Softwarekategorien nach Lehman | 48 |
| 4.1.1 | S-Typ-Systeme | 48 |
| 4.1.2 | P-Typ-Systeme | 49 |
| 4.1.3 | E-Typ-Systeme | 49 |
| 4.1.4 | Ursache für die Softwareevolution | 50 |
| 4.2 | Gesetze der Softwareevolution | 51 |
| 4.2.1 | Gesetze nach Lehmann | 51 |
| 4.2.2 | Bemerkungen zu den Gesetzen | 55 |
| 4.2.3 | Einfluss der Erstentwicklung | 57 |
| 4.3 | Lehren aus den Gesetzen der Softwareevolution | 57 |
| 4.4 | Zusammenfassung | 60 |
| Teil II Managementthemen der Softwarewartung | | 61 |
| 5 | Einbettung des Wartungsprozesses | 63 |
| 5.1 | Prozesse rund um die Wartung | 63 |
| 5.1.1 | Tätigkeitsbereiche und ihre Zielsetzungen rund um die Wartung | 63 |
| 5.1.2 | Was genau ist der Wartungsprozess? | 67 |
| 5.2 | Beiträge der verschiedenen Tätigkeitsbereiche zur Wartung | 69 |
| 5.2.1 | Die Lebenslinie der (Weiter-)Entwicklung | 69 |
| 5.2.2 | Die Lebenslinie des speziellen Wartungsprozesses | 71 |
| 5.2.3 | Die Lebenslinie des Supportprozesses | 73 |
| 5.3 | Der Produktlinienmanagement-Prozess | 76 |
| 5.3.1 | Grundgerüst eines allgemeinen Wartungsprozesses | 76 |
| 5.3.2 | Änderungsmanagement | 79 |
| 5.3.3 | Paketierung | 83 |
| 5.4 | Zusammenfassung | 85 |

| | | |
|----------|--|------------|
| 6 | Der Wartungsprozess | 87 |
| 6.1 | Der spezielle Wartungsprozess und seine Ablauforganisation | 87 |
| 6.2 | Aufbauorganisation und Rollen | 97 |
| 6.2.1 | Rollen in der Wartung | 98 |
| 6.2.2 | Aufbauorganisation – permanente Organisation oder Projektorganisation? | 102 |
| 6.2.3 | Formen der Aufbauorganisation | 103 |
| 6.3 | Konfigurationsmanagement in der Wartung | 107 |
| 6.3.1 | Zweck und Begriffe des Konfigurationsmanagements | 107 |
| 6.3.2 | Konfigurationsmanagement im Kontext der Softwarewartung | 109 |
| 6.4 | Qualitätsmanagement in der Wartung | 111 |
| 6.4.1 | Was ist Qualität? | 111 |
| 6.4.2 | Regelkreis der Qualitätssicherung | 113 |
| 6.4.3 | Auswahl der Maßnahmen in der Wartung | 117 |
| 6.5 | Dokumentation in der Wartung | 119 |
| 6.6 | Zusammenfassung | 120 |
| 7 | Wissen – Bedeutung in der Wartung | 123 |
| 7.1 | Wissen aufbauen heißt Mitarbeiter aufbauen | 123 |
| 7.1.1 | Voraussetzungen schaffen | 125 |
| 7.1.2 | Vorbilder gewinnen | 125 |
| 7.1.3 | Nachhaltigkeit sicherstellen | 128 |
| 7.2 | Umgang mit Wissen | 129 |
| 7.2.1 | Wert von Wissen | 130 |
| 7.2.2 | Wissensinventar | 131 |
| 7.2.3 | Wissensaufbau und -transfer | 137 |
| 7.3 | Retrospektive | 141 |
| 7.3.1 | Themen und Teilnehmerkreis | 142 |
| 7.3.2 | Ablauf einer Retrospektive | 143 |
| 7.4 | Zusammenfassung | 145 |
| 8 | Wartungskosten | 147 |
| 8.1 | Finanzierungsmöglichkeiten für die Wartung | 148 |
| 8.2 | Kostenblöcke in der Wartung | 151 |

- 8.3 Vorhersage der Wartungskosten 154
 - 8.3.1 Entwicklungs- versus Wartungsaufwand 154
 - 8.3.2 Kalkulation der Wartungskosten 155
 - 8.3.3 Wartungsroadmap 158
- 8.4 Wartungskosten positiv beeinflussen 165
 - 8.4.1 Gemeinsames Verständnis 166
 - 8.4.2 Schrittweises Vorgehen 167
 - 8.4.3 Folgekosten aus der Entwicklung 168
 - 8.4.4 Übergang Entwicklung – Wartung 169
 - 8.4.5 It’s not a bug, it’s a feature 170
 - 8.4.6 Einfluss von COTS 171
- 8.5 Outsourcing und Offshoring in der Wartung 173
 - 8.5.1 Begriffe 173
 - 8.5.2 Outsourcing und Offshoring allgemein 174
 - 8.5.3 Outsourcing und Offshoring im Kontext der Wartung 176
- 8.6 Zusammenfassung 178

Teil III Techniken der Softwarewartung 181

- 9 Metriken 183**
 - 9.1 Was sind Metriken? 184
 - 9.1.1 Was wir vermessen können 186
 - 9.1.2 Wie wir das richtige Maß finden 187
 - 9.1.3 Was ein gutes Maß ausmacht 189
 - 9.2 Prozessmetriken: Prozesse verbessern 190
 - 9.2.1 Prozessverbesserungen erreichen 190
 - 9.2.2 Beispiele von Prozessmetriken 191
 - 9.3 Produktmetriken: Produktzerfall im Fokus 194
 - 9.3.1 Größen-/Volumenmetriken (konventionell) 197
 - 9.3.2 Struktur-/Komplexitätsmetriken (konventionell) 198
 - 9.3.3 Größen-/Volumenmetriken (objektorientiert) 202
 - 9.3.4 Struktur-/Komplexitätsmetriken (objektorientiert) 203
 - 9.4 Ein Kompass für die Wartung 205
 - 9.5 Zusammenfassung 207

| | | |
|-----------|---|------------|
| 10 | Softwareanalyse und -visualisierung | 209 |
| 10.1 | Programmverstehen | 210 |
| 10.1.1 | Ziel des Programmverstehens | 210 |
| 10.1.2 | Artefakte als Basis zum Programmverstehen | 211 |
| 10.2 | Softwarevisualisierung | 212 |
| 10.2.1 | Möglichkeiten und Nutzen | 212 |
| 10.2.2 | Beispiel: Polymetrische Sichten | 213 |
| 10.3 | Softwareanalysetechniken | 215 |
| 10.3.1 | Statische Analyse | 215 |
| 10.3.2 | Dynamische Analyse | 215 |
| 10.3.3 | Evolutionäre Analyse | 216 |
| 10.4 | Architekturanalyse | 216 |
| 10.4.1 | Funktionsweise einer Architekturanalyse | 217 |
| 10.4.2 | Beispiele statischer Architekturanalysen | 218 |
| 10.5 | Codeanalyse | 222 |
| 10.5.1 | Beispiele statischer Codeanalysen | 222 |
| 10.5.2 | Beispiele dynamischer Codeanalysen | 228 |
| 10.5.3 | Beispiele evolutionärer Codeanalysen | 229 |
| 10.6 | Werkzeugunterstützung | 230 |
| 10.7 | Zusammenfassung | 231 |
| 11 | Reengineering, Sanierung und Migration | 233 |
| 11.1 | Begriffswelt des Reengineering | 234 |
| 11.1.1 | Reverse Engineering | 234 |
| 11.1.2 | Restrukturierung | 235 |
| 11.1.3 | Forward Engineering | 235 |
| 11.1.4 | Reengineering | 236 |
| 11.2 | Verbesserungen der Wartbarkeit mit Refactoring | 237 |
| 11.2.1 | Wann wird üblicherweise refactored? | 237 |
| 11.2.2 | Tests als Sicherheitsnetz | 237 |
| 11.2.3 | Exemplarische Bad Smells, Refactorings und Transformationen | 238 |
| 11.2.4 | Ausführungen zu Bad Smells und Refactorings | 240 |
| 11.2.5 | Refactorings sind sinnvoll, sind sie aber bezahlbar? | 242 |
| 11.3 | Sanierung: Wenn Software ernsthaft krank ist | 243 |
| 11.4 | Migration: Schritt halten mit der Umgebung | 245 |
| 11.5 | Zusammenfassung | 247 |

| | | |
|--|--|------------|
| 12 | Test..... | 249 |
| 12.1 | Grundlagen des Testens | 249 |
| 12.1.1 | Fehlerbegriff | 250 |
| 12.1.2 | Ziel des Testens..... | 251 |
| 12.1.3 | Validierung versus Verifizierung | 252 |
| 12.1.4 | Testaufwand versus Produktqualität | 253 |
| 12.2 | Testen in der Wartung | 254 |
| 12.2.1 | Die Auswirkungsanalyse als zentrales Element | 254 |
| 12.2.2 | Testaufwand im Wartungsfall reduzieren | 258 |
| 12.3 | Testmethoden..... | 260 |
| 12.3.1 | Klassifikation der Testmethoden..... | 260 |
| 12.3.2 | Statische Tests..... | 263 |
| 12.3.3 | Dynamische Tests | 269 |
| 12.3.4 | Wichtige Testmethoden in der Wartung | 280 |
| 12.4 | Zusammenfassung | 282 |
| Anhang | | 285 |
| Wartungsmanagement kompakt..... | | 287 |
| Abkürzungen | | 289 |
| Glossar | | 291 |
| Literatur..... | | 303 |
| Webreferenzen | | 309 |
| Index | | 311 |