

Inhaltsverzeichnis

- 1 Thesen..... 1**
- 1.1 Einführung in das Konzept..... 1
- 1.2 Mehr Chancen durch bessere Strategien..... 5
- 1.2.1 Optimale Arbeitsteilung 7
- 1.2.2 Arbeitsplätze im Wandel 9
- 1.2.3 Synergie muss organisiert werden 10
- 1.2.4 Offen sein für Problemlösungen..... 13
- 1.2.5 Helfen, nicht beschränken 14
- 1.2.6 Probleme richtig verstehen 15
- 1.2.7 Simplifizieren durch Organisieren..... 16
- 1.2.8 Mehr erreichen durch strategische Entscheidungen 21
- 1.2.9 Interdisziplinäre Kooperation – eine effektive Strategie 23
- 1.2.10 Mehr Zuverlässigkeit und Effizienz durch Automation 27
- 1.2.11 Ohne richtige Dimensionierung geht nichts 29
- 1.2.12 Komplexität – weniger ist mehr 30
- 1.2.13 Mensch-Maschine-Schnittstelle..... 33
- 1.2.14 Ohne Zusatzaufwand unendlich viele Fälle abdecken 36
- 1.2.15 Qualitätssicherung 37
- 1.2.16 Schnittstellenanpassung 40
- 1.2.17 Dokumentation 41
- 1.3 Thesen zur effizienten Softwareentwicklung 42
- 1.3.1 Stand der Technik..... 42
- 1.3.2 Wettbewerb..... 42
- 1.3.3 Softwareentwicklung..... 42
- 1.3.4 Organisation 43
- 1.3.5 Schnittstellen 44
- 1.3.6 Information 45
- 1.3.7 Automation 45
- 1.3.8 Strategie..... 46
- 1.3.9 Kostenschätzung..... 47
- 1.3.10 Projektmanagement 47

| | | |
|----------|------------------------------------------------------------------------------------|-----------|
| 1.3.11 | Management | 48 |
| 1.3.12 | Qualitätssicherung | 48 |
| 2 | Strategische Ausrichtung | 51 |
| 2.1 | Wann ist man effizient und erfolgreich? | 52 |
| 2.2 | Mehr Effizienz durch Automatische Softwareproduktion ... | 53 |
| 2.3 | Software korrekt produzieren durch Automation | 55 |
| 2.3.1 | Anforderungen korrekt umsetzen | 56 |
| 2.3.2 | Verifikation und Validierung – was Softwareentwickler und Bäcker brauchen | 57 |
| 2.3.3 | Synergien erzeugen..... | 58 |
| 2.3.4 | Vom Compiler zum Systemcompiler..... | 60 |
| 2.3.5 | Selbstkontrolle | 61 |
| 2.3.6 | Ist Selbstkontrolle sinnvoll?..... | 62 |
| 2.4 | Entwickeln impliziert Projektmanagement..... | 63 |
| 2.5 | Organisation der Entwicklung | 65 |
| 2.6 | Wie groß ist das Einsparungspotenzial? | 67 |
| 2.7 | Welcher Weg ist der beste?..... | 68 |
| 2.8 | Automatisierung effizient einsetzen..... | 74 |
| 2.9 | Weniger ist mehr - durch Einschränkungen mehr erreichen | 76 |
| 2.10 | Effizient Entwicklungsrisiken meistern | 79 |
| 2.11 | Komplexität meistern..... | 82 |
| 2.12 | Effiziente Wartung..... | 86 |
| 3 | Risiken und Chancen in der Softwareentwicklung | 89 |
| 3.1 | Rollenverteilung | 89 |
| 3.2 | Professionelle Softwareentwicklung – <i>die Herausforderung</i> | 91 |
| 3.2.1 | Risiken durch Informationsmangel..... | 92 |
| 3.2.2 | Risiken durch Managementfehler | 93 |
| 3.2.3 | Der Kampf gegen das Chaos..... | 94 |
| 3.3 | Fehlentscheidungen | 96 |
| 3.3.1 | Viel Code, wenig Qualität..... | 96 |
| 3.3.2 | Wie zuverlässig sind Ergebnisse? | 97 |
| 3.3.3 | Der Anwender ist nicht König | 102 |
| 3.4 | Risiken | 108 |
| 3.4.1 | Interne Risiken..... | 108 |
| 3.4.2 | Externe Risiken..... | 116 |
| 3.5 | Softwareentwicklungswerkzeuge | 126 |
| 3.5.1 | Anwendungsprofil und Parametrisierbarkeit | 126 |
| 3.6 | Geringeres Risiko durch schnellere Umsetzung? | 129 |
| 3.7 | Durch Tests Fehler erkennen | 131 |
| 3.8 | Qualität | 137 |
| 3.9 | Organisation der Automation..... | 140 |

| | | |
|----------|--------------------------------------------------------------------------------|------------|
| 3.10 | Komplexität | 143 |
| 3.10.1 | Große Mengen, große Komplexität | 144 |
| 3.10.2 | Unübliche Regeln, hohe Komplexität..... | 144 |
| 3.10.3 | Gegenmaßnahmen | 146 |
| 3.11 | Das Potenzial der Automation | 148 |
| 4 | Entwicklungsgrundlagen | 151 |
| 4.1 | Manuelle Softwareentwicklung | 151 |
| 4.2 | Entwicklungsansätze | 152 |
| 4.3 | Anforderungen an automatische Prozesse | 158 |
| 4.3.1 | Grenzen der Teilautomation | 158 |
| 4.3.2 | Einfluss auf Korrektheit, Konsistenz und Vollständigkeit..... | 158 |
| 4.4 | Qualitätssicherung | 161 |
| 4.4.1 | Qualitätsmerkmale..... | 161 |
| 4.4.2 | Qualitätssicherungsstandards..... | 163 |
| 4.4.3 | Zertifizierung | 166 |
| 4.4.4 | Klassifizierung von Werkzeugen..... | 168 |
| 4.4.5 | Verifikation und Validierung..... | 168 |
| 4.5 | Zuverlässigkeit | 172 |
| 4.5.1 | Pragmatische Ansätze zur Bestimmung der Zuverlässigkeit | 172 |
| 4.5.2 | Aktive Qualitätssicherung: Maßnahmen gegen Fehler | 174 |
| 4.6 | Betriebssysteme – Effizienz und Risiken | 196 |
| 4.6.1 | Softwarequalität hängt vom Betriebssystem ab | 197 |
| 4.6.2 | Klassifikationskriterien..... | 198 |
| 4.7 | Hilfen für die Softwareentwicklung | 205 |
| 4.7.1 | Methoden..... | 206 |
| 4.7.2 | Generatoren | 209 |
| 4.7.3 | Sprachen | 212 |
| 4.7.4 | Unterstützung der Fehlerprävention und Fehlererkennung durch Sprachen | 218 |
| 5 | Managementaspekte..... | 227 |
| 5.1 | Das Rationalisierungspotenzial | 227 |
| 5.2 | Was lässt sich ändern? | 230 |
| 5.3 | Argumente und Gegenargumente | 232 |
| 5.3.1 | Der indifferente Ansatz | 232 |
| 5.3.2 | Der "Mengen"-Ansatz | 233 |
| 5.4 | Wo lohnt es sich?..... | 235 |
| 5.4.1 | Kosten vs. Entwicklungsphasen | 236 |
| 5.4.2 | Kosten der Fehlerbeseitigung | 240 |
| 5.4.3 | Sparen bei Wartung und Pflege / Legacy Software | 244 |
| 5.4.4 | Strategie zur Kostensenkung | 245 |

| | | |
|----------|----------------------------------------------------------------|------------|
| 5.5 | Organisatorische Maßnahmen | 248 |
| 5.5.1 | Das Baukastensystem | 249 |
| 5.5.2 | Spezifikation | 252 |
| 5.5.3 | Test, Verifikation und Validierung | 256 |
| 5.5.4 | Die Abnahme | 257 |
| 5.5.5 | Wartung und Legacy Systeme | 259 |
| 5.5.6 | Vertragliche Aspekte und Projektorganisation | 260 |
| 5.5.7 | Hindernisse wegräumen | 263 |
| 5.5.8 | Qualitätssicherung durch Standards | 266 |
| 5.6 | Wettbewerbsvorteile | 266 |
| 5.7 | Managementaufgaben | 267 |
| 5.7.1 | Kostenanalyse | 268 |
| 5.7.2 | Erfolgsanalyse | 270 |
| 5.7.3 | Risikoanalyse und Risikominimierung | 271 |
| 5.7.4 | Fehleranalysen | 273 |
| 5.7.5 | Empfehlungen zur Optimierung | 274 |
| 5.8 | Checkliste | 276 |
| 6 | Automatische Softwareproduktionsprozesse..... | 291 |
| 6.1 | Ziele des vollautomatischen Produktionsprozesses | 291 |
| 6.2 | Voraussetzungen für die Prozessdefinition | 293 |
| 6.3 | Entwicklerprofile | 295 |
| 6.4 | Die Prozessdefinition | 296 |
| 6.5 | Produktionsprozesse – im wesentlichen nichts Neues | 298 |
| 6.5.1 | Anwendungsorientierte Produktionsprozesse | 298 |
| 6.5.2 | Spezialisierung vs. Vereinheitlichung | 301 |
| 6.5.3 | Der konfigurierbare Produktionsprozess | 303 |
| 6.5.4 | Vollautomatische Produktionsprozesse | 305 |
| 6.5.5 | Prozessparameter | 306 |
| 6.5.6 | Typen von Produktionsprozessen | 312 |
| 6.5.7 | Identifizierung von Abhängigkeiten | 321 |
| 6.5.8 | Konstruktionsregeln | 323 |
| 6.5.9 | Metamodelle | 333 |
| 6.5.10 | Eigenschaften vollautomatischer Produktionsprozesse | 333 |
| 6.5.11 | Verifikation von Prozessen und Qualitätssicherung | 339 |
| 6.5.12 | Validierung und automatische Testfallerzeugung | 348 |
| 6.5.13 | Kontinuierliche Wartung | 349 |
| 6.5.14 | Systemcompiler | 350 |
| 6.5.15 | Teilprodukte und vertragliche Aspekte | 351 |
| 6.6 | Der automatische Entwicklungszyklus | 356 |
| 6.7 | Die Vorteile für ein einzelnes Projekt | 359 |
| 6.8 | Merkmale eines vollautomatischen Produktionsprozesses | 361 |
| 6.9 | Weitere Ansätze auf dem Gebiet der Automation | 366 |

| | | |
|----------|------------------------------------------------------------|------------|
| 7 | Vollautomation als Realität | 369 |
| 7.1 | Über die Einführung und Optimierung neuer Methoden... | 369 |
| 7.2 | Durchgängig von der Spezifikation bis zum Betrieb | 373 |
| 7.2.1 | Von der Spezifikation zum Produkt | 374 |
| 7.2.2 | Die Realisierung | 382 |
| 7.2.3 | Ergebnisse | 384 |
| 7.3 | Reengineering und n-Version-Development | 388 |
| 7.3.1 | Vom Reengineering zu n-Version-Development..... | 391 |
| 7.4 | Redundanzreduktion bei Parsern | 395 |
| 7.4.1 | Formale Beschreibung von Sprachen | 396 |
| 7.4.2 | Transformation von Grammatiken | 397 |
| 7.4.3 | Die Entstehung von Redundanzen..... | 399 |
| 7.4.4 | Die Risiken von Redundanzen und ihre Eliminierung | 400 |
| 7.5 | Algorithmen..... | 401 |
| 7.5.1 | Datenkonvertierung | 402 |
| 7.5.2 | Ausnutzung theoretischer Möglichkeiten in der Praxis | 407 |
| 7.6 | Verteilte Echtzeitsysteme | 409 |
| 7.6.1 | Master-Slave Struktur..... | 412 |
| 7.6.2 | Echtzeitinfrastuktur | 416 |
| 7.6.3 | Spezifikation und Erzeugung der MSL-Datenbank | 420 |
| 7.7 | Ein synchrones verteiltes System | 423 |
| 7.8 | Client-Server Anwendungen, Datenbanken und GUIs | 425 |
| 7.8.1 | SQL-Datenbanken aus einem Guss | 430 |
| 7.8.2 | Grafische Oberflächen aus C-Typdefinitionen | 437 |
| 7.9 | Schnittstellenanpassung..... | 439 |
| 7.10 | Automatische Testumgebungen..... | 440 |
| 7.10.1 | Funktionstests..... | 440 |
| 7.10.2 | Test des Verhaltens..... | 442 |
| 7.11 | Effizienzanalyse | 443 |
| 7.12 | Rückblick und Ausblick | 449 |
| 8 | Gesellschaftliche Aspekte..... | 451 |
| 8.1 | Sozialpolitische Relevanz..... | 452 |
| 8.1.1 | Innovation und die Folgen..... | 453 |
| 8.1.2 | Sozialverträglichkeit..... | 454 |
| 8.1.3 | Konkurrenzfähigkeit..... | 455 |
| 8.1.4 | Der Softwareentwickler, zuerst geschützt, dann gehetzt ... | 456 |
| 8.2 | Das Arbeitsumfeld..... | 458 |
| 8.2.1 | Rückblick..... | 458 |
| 8.2.2 | Ausblick..... | 459 |
| 8.2.3 | Kooperation ist notwendig..... | 460 |
| 8.2.4 | Der Anpassungsprozess..... | 460 |

| | | |
|----------|----------------------------------------------------|------------|
| 8.3 | Wirtschaftliche Relevanz..... | 464 |
| 8.3.1 | Mehr Synergie durch neue Produktionsmethoden | 464 |
| 8.3.2 | Egalisieren vs. Spezialisieren..... | 466 |
| 8.3.3 | Automatische Produktion vs. Fremdentwicklung | 468 |
| 8.4 | Ausbildungspolitische Relevanz | 469 |
| 8.4.1 | Der Kunde ist König..... | 469 |
| 8.4.2 | Automation impliziert Qualifikation..... | 471 |
| 8.5 | Zusammenfassung | 473 |
| 9 | Schlussbemerkungen und Ausblick..... | 475 |
| | Abkürzungen und Synonyme | 479 |
| | Begriffe | 483 |
| | Literatur | 499 |
| | Sachverzeichnis..... | 511 |

111 Thesen zur erfolgreichen Softwareentwicklung

Argumente und Entscheidungshilfen für Manager.

Konzepte und Anleitungen für Praktiker

Gerlich, R.

2005, XXII, 522 S., Hardcover

ISBN: 978-3-540-20910-2