

Hausaufgabe 4

Tim Wende

29. Oktober 2021

1 Softwarekrise

Die Softwarekrise wurde als das Problem eingeführt, das mit Softwaretechnik gelöst werden soll. Auch wenn der Begriff aus den 60er Jahren stammt, so ist er auch heute noch anwendbar.

- a. Wählen Sie einen charakteristischen Aspekt der Softwarekrise aus der Vorlesung (oder externen Quellen) und erläutern Sie ihn in eigenen Worten. Zwei Sätze genügen.

Komplexe, nicht mathematisch beschreibbare Programme für "einfache" Aufgaben:

Auf Grund starken Veränderungen im Produktivsystem im Vergleich zu Testumgebungen müssen unendlich viele mögliche Sonderfälle beim Erstellen von Software betrachtet werden. Dadurch werden selbst kleinst-Programme derart aufgeblasen, dass sie deutlich komplexer sind, als sie eigentlich sein müssten. Beispielsweise, wenn verschiedene Betriebssysteme in config files berücksichtigt werden müssen (Beim entpacken von .zip, .tar etc).

- b. Die Charakterisierung der Softwarekrise spricht verschiedene Ebenen an. Erläutern Sie die Problematik bzgl. **Qualitätssicherung** und **Ökonomie** (Entwicklungsaufwand)

Die momentane reine Entwicklungszeit von Programmen ist nur ein Bruchteil des benötigten Arbeitsaufwands. Hinzu kommen komplexe, zeitintensive Tests sowie andauernde Betreuung dieser Programme. Vor allem die Betreuung auf unbestimmte Zeit kostet Zeit sowie Humankapital. Als alternative kann jegliche Software für verschiedene Systeme immer wieder neu programmiert werden. Dies ist jedoch weder ökonomisch, noch Sinnvoll im Bezug auf Wirtschaftlichkeit der Unternehmen. Unter Berücksichtigung dieser Punkte sollte jedoch nicht die Qualität der Produkte leiden. Diese sollen weiterhin in Design, Funktionalität sowie Wartbarkeit die Anforderungen erfüllen.

- c. Nennen Sie eine Erfahrung, die die Auffassung bestärkt oder widerlegt, dass die Softwarekrise noch anhält.

WINDOWS XP SERVER AN DER RWTH! Auf Grund veralteter Software und mangelnder Kompatibilität zu zugehörigen Produkten aus diesem Jahrtausend, zieht diese Software die Qualität und Upgrade-Bereitschaft anderer Programme runter. So wird mittelalterliche Software, meist auf virtualisierten Rechnern, weiterhin zur Verfügung gestellt und dient als einfacher Einstieg, wenn man solch einem Unternehmen z.B. durch Schadcode schaden möchte.

2 Charakterisierung Softwaretechnik

In der Vorlesung wurde eine zusammenfassende Definition von Softwaretechnik vorgestellt.

- a. Welche softwaretechnischen Mittel werden in Softwareprojekten eingesetzt?

„Zielorientierte Bereitstellung und systematische Verwendung von Prinzipien, Methoden und Werkzeugen für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Softwaresystemen.“

– Lit.: Balzert, S. 36

Zu den Kernprozessen von Softwaretechnik gehören:

1. **Planung**
Anforderungsanalyse, Lastenheft, Pflichtenheft, Aufwandsschätzung, Vorgehensmodell
2. **Analyse**
Auswertung, Mock-Up, Prozessanalyse, Systemanalyse, SA, OOA
3. **Entwurf**
Softwarearchitektur, SD, OOD, FMC
4. **Programmierung**
Normierte Prog., Strukturierte Prog., OOP, Funktionale Prog.
5. **Validierung und Verifikation**
Modultests, Integrationstests, Systemtests, Akzeptanztests

Zu den Unterstützungsprozessen von Softwaretechnik gehören:

6. **Anforderungsmanagement**
7. **Projektmanagement**
Risikomanagement, Projektplanung, Projektverfolgung, Lieferantenvereinbarungen
8. **Qualitätsmanagement**
SPICE, Incident Management, Problemmanagement, Softwaremetrik, statische Analyse
9. **Konfigurationsmanagement**
Versionsverwaltung, Änderungsmanagement, Releasemanagement, ITIL
10. **Softwareeinführung**
11. **Dokumentation**
Technische Dokumentation, Softwaredokumentation, Bedienungsdokumentation

- b. Welche Eigenschaften von Software-Produkten sollen dadurch erreicht werden?

Durch exakte Planung, daraus resultierender Konzepte und genauer Betrachtung der Anforderungen sollen Eigenschaften der Qualität verbessert werden. Des Weiteren sollen durch anschließende Tests die Anzahl an möglichen Fehlern (falls möglich auf 0 [*Spoiler-Alarm, ist nie möglich*]) gesenkt werden. So soll an den Käufer ein vollständiges, den Anforderungen entsprechendes, möglichst fehlerfreies, effizientes, wartbares und zuverlässiges Produkt geliefert werden.

- c. Wodurch unterscheidet sich Softwaretechnik damit von handwerklich ordentlichem Programmieren?

Stumpf gesagt wirkt sich die korrekte Anwendung sinnvoller Softwaretechnik auf die Planung von Software aus. So wird sich vor dem Programmieren bei SWT Gedanken über mögliche Fehlerquellen gemacht, und bei dem handwerklich ordentlichen Programmieren erst im Fehlerfall. Somit ist das stumpfe Programmieren ein kleiner Teilbereich von Softwaretechnik, verpackt mit Planung, Konzeptionierung, sowie anschließender Evaluierung.

3 Git

Im Praktikum wurde das Versionsverwaltungstool Git vorgestellt. Beantworten Sie die folgenden Aufgaben:

- a. Was sind die Vorteile bei der Verwendung von Git im Gegensatz zum Arbeiten ohne Versionsverwaltungstools?

Bei eventuell auftretenden Fehlern oder ungewollten „Features“ kann auf eine ältere Version des Codes zurückgespielt werden. Des Weiteren kann man sich ältere Versionsstände anschauen, sowie Updates genauer inspizieren. So sieht man bei einem sogenannten `commit` meist einen beschreibenden Updategrund und kann im Fehlerfall den schuldigen identifizieren.

- b. Wie unterscheidet sich Git von einem zentralisierten VCS wie zum Beispiel Subversion und erläutern Sie die Vorteile, die sich aus diesen Unterschieden ergeben?

Git funktioniert ebenfalls lokal auf dem eigenen System. Des Weiteren ist git schneller und das Zusammenfügen verschiedener Branches ist Benutzerfreundlicher.

- c. Nehmen Sie an, Sie arbeiten an einem Softwareprojekt mit vorhandenen Commits. Sie schreiben Code für eine bereits existierende Klasse und stellen im Nachhinein fest, dass Ihre Änderungen das Problem nicht lösen. Wie lassen sich mit Git Ihre getätigten Änderungen rückgängig machen? Unterscheiden Sie dabei zwischen:

- noch nicht geadded
`strg+z`
- schon geadded
`git checkout`
- schon commitet
`git reset`