

# Software-Management

# Software-Management

Das **Software-Management** besteht im wesentlichen aus den drei Bereichen:

- **Projektmanagement**
- **Konfigurationsmanagement**
- **Qualitätsmanagement**

# Projekt-Management

## Projektmanagement

Der Projektmanager muss über Daten von hoher Aktualität verfügen, wie z.B.:

- Plandaten:** als Übersicht über Entwicklungsstand, aktuell einsetzbarer Mittel und Ressourcen (inkl. Personal) in jeder Phase der Produktentwicklung.
- Risikoanalysen:** als quantitative Aussagen zu Entwicklungsvarianten und Entwicklungsproblemen mit damit zusammenhängenden Risiken,
- Kostenanalysen:** auf die Kosten bezogene Angaben zum Entwicklungsstand, den noch vorhandenen und den aktuell noch benötigten Mitteln,
- Reviews und Audits:** als Fehleranalysen, Qualitätsbeschreibungen und Prüfberichte.

## Projektmanagement

Weitere Hilfsmittel und Begriffe für das Projektmanagement:

**Meilensteine:** (*milestone*) Planungsabschnitt, für den exakte Zielangaben existieren. Wird vor allem zur Terminplanung eingesetzt.

**Function-Point-Methode :** Methode zur direkten und indirekten Aufwands- und Kostenabschätzung

**Personalplanung :** Planung des Einsatzes des verfügbaren Personals.  
**Wichtige Faktoren:**  
Projektumfang, Projektplan (Zeitplan),  
Personal (Verfügbarkeit, Ausbildungsstand und Erfahrung),  
(daraus:) Kosten des Personals,  
Motivation des Personals,....

# Konfigurations-Management

## Konfigurationsmanagement

Es dient zur Sicherstellung der Konsistenz der Verknüpfungen zwischen Programmen, Objekten, Modulen, Versionen, etc. eines (SW-)Produkts.

Eine Konsistenzproblematik macht sich spätestens dann bemerkbar, wenn die Beteiligten sich ärgern, dass

- Sie die eben bearbeiteten Programmversionen verändert wiederfinden
- die letzte Version nicht mehr auffindbar ist
- Programme, die früher fehlerfrei liefen, nach einer Neukompilierung plötzlich nicht mehr lauffähig sind
- Änderungen nicht in allen Softwarekomponenten nachgeführt wurden
- nicht alle zu einem Produkt gehörenden Komponenten auffindbar sind
- wiederverwendbare Komponenten mehrfach programmiert wurden
- keine Transparenz über die eingesetzten Komponenten besteht

# Konfigurationsmanagement

Das **Konfigurationsmanagement** stellt sicher, dass

- Produkte eindeutig identifizierbar sind,
- Zusammenhänge und Unterschiede von verschiedenen Versionen einer Konfiguration erkennbar bleiben und
- Produktänderungen nur kontrolliert durchgeführt werden können

Es kann in vier Bereiche untergliedert werden (s. V-Modell):

- KM-Planung
- Produkt- und Konfigurationsverwaltung
- Änderungsmanagement (KM-Steuerung)
- KM-Dienste



# Konfigurationsmanagement

## Konfigurationsmanagement-Planung

- Verbindliche Festlegung der für das Projekt geltenden **Richtlinien** und **Verfahren**  
→ Voraussetzung für ein zuverlässiges Konfigurationsmanagement

## Produkt- und Konfigurationsverwaltung

- stellt sicher, dass Produkte und Konfigurationen **eindeutig identifiziert**, **zugriffsgesichert** und **rekonstruierbar** gespeichert sind.
  1. Einsatz einer Versionsverwaltung (CVS, subversion, SourceSafe, ClearCase, Git, ... )  
→ muss geplant und konsequent durchgeführt werden!
  2. Zugriff auf die Konfigurationselemente soll **gesteuert** erfolgen,  
→ Produkte (Module) können weder absichtlich noch unabsichtlich zerstört werden.
  3. Zur Rekonstruierbarkeit: sämtliche Änderungen am System **zurückverfolgbar** gestalten  
→ setzt die Identifizierbarkeit aller Produkte voraus!
  4. Automatisierte Konfigurationserstellung → CI (Continuous Integration)
  5. Automatisierte Auslieferung der Software → CD (Continuous Delivery)

## Konfigurationsmanagement-Dienste

Sie werden in bestimmten Intervallen oder nach Bedarf durchgeführt.

### Zu den KM-Diensten zählen:

- *Daten administrieren* Gewährleistung eines **zentralen**, **unternehmensweiten**, einheitlichen und konsistenten Datenkatalogs
- *SW-Produkte katalogisieren* dadurch Wiederverwendbarkeit ermöglichen
- *Schnittstellen koordinieren* Kompatibilität und Vermeidung einseitiger Änderungen
- *Ergebnisse sichern* zur Dokumentation des erreichten Projektstands und um sie für spätere Projekte zur Verfügung zu stellen

### Weitere KM-Dienste:

- *KM-Dokumentation führen* Dokumenterstellung für KM-Belange
- *Release-Management* Kontrolle der Konfigurationsfreigabe und -verteilung
- *Projekthistorie führen* Gewährleistung einer umfassenden und nachvollziehbaren Dokumentation des Projektverlaufs zur **Verbesserung künftiger Projekte**

## Release-Management als KM-Dienst

- Planung und Festlegung, **wann welche** Version (Release) des Produkts **mit welchen Inhalten** fertig gestellt wird.
- Zwei Aufgabenbereiche:
  - 1. Release-Planung**

Vorteile:

    - Arbeit im Projekt besser planbar
    - Fortschrittsmessung anhand der erzielten Ergebnisse möglich
    - Verwaltung unterschiedlicher Varianten eines Produkts für verschiedene Kunden
  - 2. Verwaltung der verschiedenen Stände**
    - Verwendung eines Versionsverwaltungssystems (z. B. CVS, subversion, SourceSafe, ClearCase, Git, ... ).

## Konfigurations-Arten

### Vollproduktkonfiguration

eine Konfiguration über ein **ganzes Produkt** hinweg. Es wird eine **komplette Stücklistenauflösung** mittels der Produktwurzeln vorgenommen

### Teilproduktkonfiguration

beziehen sich nur **auf Teile eines Produktes**. Andere Elemente von anderen Produkten werden nicht mit einbezogen. Es wird ebenfalls eine Stücklistenauflösung mittels der Produktwurzeln vorgenommen.

### Änderungskonfiguration

Nur wenn Änderungen ausgeliefert werden. Eine Auflösung wird nicht vorgenommen, und es werden nur die **geänderten Module** in die Konfiguration mit einbezogen.

# Qualitäts-Management

## Was ist Qualität?

- **Raumfahrt:** Softwarefehler waren in den 90er Jahren häufigste Fehlerquelle



Explosion der ersten Ariane-5-Rakete am 4. Juni 1996

### 1996, Ariane 5:

- Software sollte von Ariane 4 übernommen werden (Kosten)
- Messwerte wurden von 64-bit Double in signed 16-bit Integer umgewandelt (Wert > 32768 → Overflow → Sprengung)

Quelle: golem.de vom 24.11.2015

## Was ist Qualität?

- **Luftfahrt:** Softwarefehler als Absturzursache



(Quellen: spiegel.de vom 19.05.2015)



(welt.de vom 28.06.2015)

- 2015, Airbus M400 • Software für Triebwerke war nicht korrekt aufgespielt (nur bei drei Triebwerken)
- Elektronik schaltete in Leerlauf

## Was ist (SW-)Qualität?

- SW-Qualität:
  - Normale Software: 10 - 20 Fehler pro 1000 Codezeilen
  - Gute Software: 2 Fehler pro 1000 Codezeilen
  - Space Shuttle: < 1 Fehler pro 10.000 Codezeilen (insg. ca. 400.000 CZ)
- Codezeilen zum Vergleich (Quelle: t3n.de, 25.05.2014):
  - Mars Curiosity Rover: 4 Mio.
  - Linux Kernel 2.6: 4 Mio.
  - Windows 7: 40 Mio (Vista 50 Mio)
  - Facebook (incl. Backend-Code): 61 Mio
  - OS X „Tiger“: 84 Mio
  - Debian 5.0 Codebase: 67 Mio.
  - Fahrzeug-SW: 100 Mio.



# Qualitäts-Management

## Qualitätsplanung

Auswählen, Klassifizieren und Gewichten der **Qualitätsmerkmale** sowie Festlegen der **Qualitätsanforderung** unter Berücksichtigung der Realisierungsmöglichkeiten

## Qualitätsprüfung

Feststellen, inwieweit eine Einheit die **Qualitätsanforderung** erfüllt

## Qualitätslenkung

Überwachung und Korrektur der Realisierung einer Einheit mit dem Ziel, die **Qualitätsanforderung zu erfüllen**.

- **Qualitätsgestaltung**  
alle Tätigkeiten zur konstruktiven Erzeugung der geforderten Qualität
- **Qualitätsbeurteilung**  
Überwachung der Qualitätsmanagement-Maßnahmen, Auswertung von Qualitätsdaten und Entscheidung über weiteres Vorgehen (z.B. Freigaben, Korrektive Maßnahmen)

## Qualitäts-Management (Prinzipien)

**Prinzipien,** abgeleitet aus den Zielen der Softwareerstellung  
(Termintreue, Kostenoptimierung und zufriedenstellende Qualität)

- Produktabhängige Qualitätsplanung
- Rückkopplung der Ergebnisse der Qualitätsprüfung
- Maximale konstruktive Maßnahmen zum Qualitätsmanagement
- Frühzeitige Fehlererkennung
- Werkzeugunterstützte Qualitätssicherung
- Integriertes entwicklungsbegleitendes Qualitätsmanagement
- Unabhängigkeit der Qualitätsprüfungen
- Bewertung von QM-Maßnahmen (interne Qualitätsaudits)

## Qualitäts-Sicherung (ISO 9000)

Zur Sicherstellung der gewünschten Produktqualität (**Produkthaftungsgesetz**)

### Definitionen:

- Qualitätssicherung:** Zusammenfassung von geplanten und systematisch durchgeführten Tätigkeiten, damit ein Produkt den vorgegebenen Qualitätsanforderungen entspricht.
- Konstruktive QS:** Verwendung von Methoden, Sprachen und Werkzeugen, die dafür sorgen, dass das entstehende Produkt à priori bestimmte Eigenschaften besitzt.
- analytische QS:** Diagnostische Maßnahmen (sie bringen in das Produkt keine Qualität per se). Messung des existierenden Qualitätsniveaus.
- Qualitätsanforderungen:** Qualitative und quantitative Eigenschaften, die an die Qualität eines Produktes gestellt werden.

## Literatur

1. Helmut Balzert,  
Lehrbuch der Softwaretechnik Band 2, Spektrum Akademischer  
Verlag, Heidelberg, Berlin, 1998, ISBN 3-8274-0065-1
2. <http://www.software-kompetenz.de/>,  
vom BMBF geförderte Web-Seiten zum Thema SW-Engineering

## Aufgabe zu PM (Erstellung eines Projektplans, Teil 1)

Für die Entwicklung eines Softwareprojekts soll ein Projektplan erstellt und im weiteren Verlauf des Projekts angepasst werden. **Randbedingungen:**

1. Alle Mitarbeiter sind gleich qualifiziert
2. SW-Auftrag entspricht bereits durchgeführten Projekten
3. Eine Arbeitswoche umfasst 5 Arbeitstage à 8 Arbeitsstunden
4. Es sind 4 Personen im Entwicklungsteam
5. Start des Projekts in 4 Wochen (von heute aus gerechnet)
6. Es laufen noch weitere Projekte:
  - Für 1 Person noch 8 Wochen (von heute aus gerechnet)
  - Für 2 Personen noch 5 Wochen
  - Für 1 Person noch 4 Wochen
7. Veranschlagte Man-Power sind 100 Personenwochen
8. Entwicklungskosten betragen 100 € pro Stunde und Mitarbeiter, an Wochenenden und bei Überstunden 150 €
9. „Überstunden“: Es dürfen max. 2 Mitarbeiter an max. 3 aufeinander folgenden Wochen maximal 6 Tage arbeiten, ansonsten nur 5 Tage. Maximale Arbeitstagdauer ist 12 Stunden.

## Aufgabe zu PM (Erstellung eines Projektplans, Teil 2)

**Aufgabe 1: Erstellen Sie einen Projektplan für obigen Auftrag (gerechnet ab kommender Woche).**

1. Welche Kosten werden für diesen Auftrag veranschlagt?
2. Welchen Fertigstellungstermin können Sie anbieten?



Start des Projekts

	Wo 1	Wo 2	Wo 3	Wo 4	Wo 5	Wo 6	Wo 7	Wo 8	...
Person 1									
Person 2									
Person 3									
Person 4									

## Aufgabe zu PM (Erstellung eines Projektplans, Teil 3)

### Aufgabe 2: Passen Sie den Projektplan an folgende Ereignisse an:

1. Mitarbeiter Nr. 2 wird in der 3. Woche nach dem Start für 2 Wochen krank.
2. Ein Mitarbeiter (Nr. 4) erinnert Sie daran, dass er in der 6. Woche nach dem Start für 3 Wochen in Urlaub ist.
3. Zur Bewältigung neuer Anforderungen wird eine neue SW-Version erforderlich. Dies schlägt mit 20.000 € zu Buche.
4. Eine weitere neue Anforderung in der 10. Woche nach dem Start verursacht eine 1-wöchige Schulung eines Mitarbeiters (10.000 € Schulungskosten).
5. Ein neuer Auftrag über 50 Personenwochen soll in 4 Wochen begonnen werden. Diese Info erschien in der 10. Woche nach dem Start (→Woche 14).
6. Eine Woche Aufschub wird durch den Kunden ermöglicht.
7. Ein Mitarbeiter (Nr. 3) wird 5 Arbeitstage früher fertig als geplant.
8. Testergebnisse erfordern eine Überarbeitung => 10 Personentage mehr!

Welche reale Kosten haben sich nun ergeben und welcher Fertigstellungstermin kann eingehalten werden?