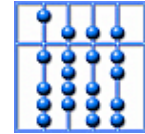


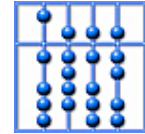
Vorlesung Projektmanagement und Teamorganisation

Dr. Bernhard Schätz
Leopold-Franzens Universität Innsbruck
Sommersemester 2003



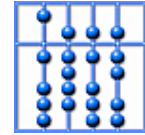
Übersicht

1. Übersicht
2. Projektmanagement und Software-Engineering
3. Projektstrukturen und Personalaktivitäten
4. Projektvorbereitung und Projektplanung
5. Projektkontrolle und Projektsteuerung
 1. Fortschrittskontrolle
 2. Risikomanagement
 3. Qualitätsmanagement (QS-Verfahren, Qualitätsmaße, Fehlermodelle)
 4. Konfigurations- und Versionsmanagement
 5. Werkzeuge
6. Projektabschluss und Prozessverbesserung
7. Ausblick: Der Faktor Mensch im Projektmanagement

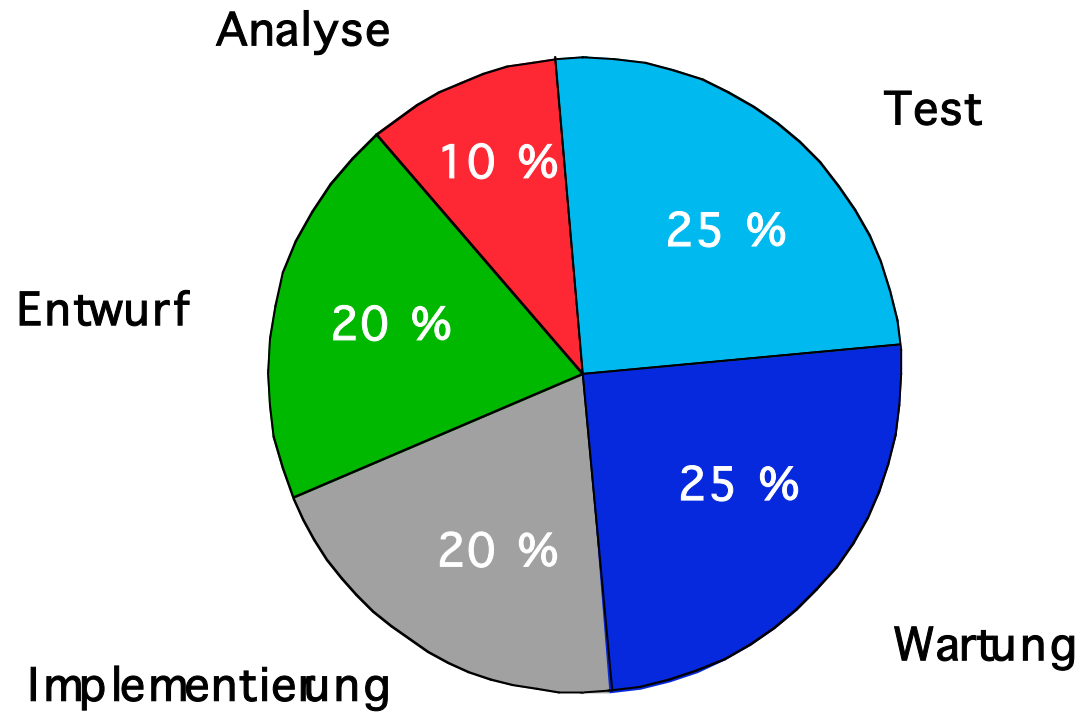


Konfigurationsmanagement

- Randbedingungen:
 - Produktkomplexität: Viele Module, viele Bearbeiter
 - Produktvielfalt: Viele unterschiedliche ausgelieferte Versionen
- Probleme:
 - Rücknahme kontraproduktiver Änderungen: Welche Version muss wiederhergestellt werden?
 - Identifikation von Fehlern: Welche (Modul-)Versionen sind davon betroffen?
 - Übernahme von Verbesserungen: Welche (Kunden-) Versionen sind davon betroffen?
 - Einführung neuer Versionen: Wo, warum, und von wem wurden Änderungen vorgenommen?
 - Kontrolle: Welche Änderungen sind durchgeführt

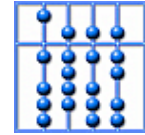


Wartungsaufwand Einzelprojekt

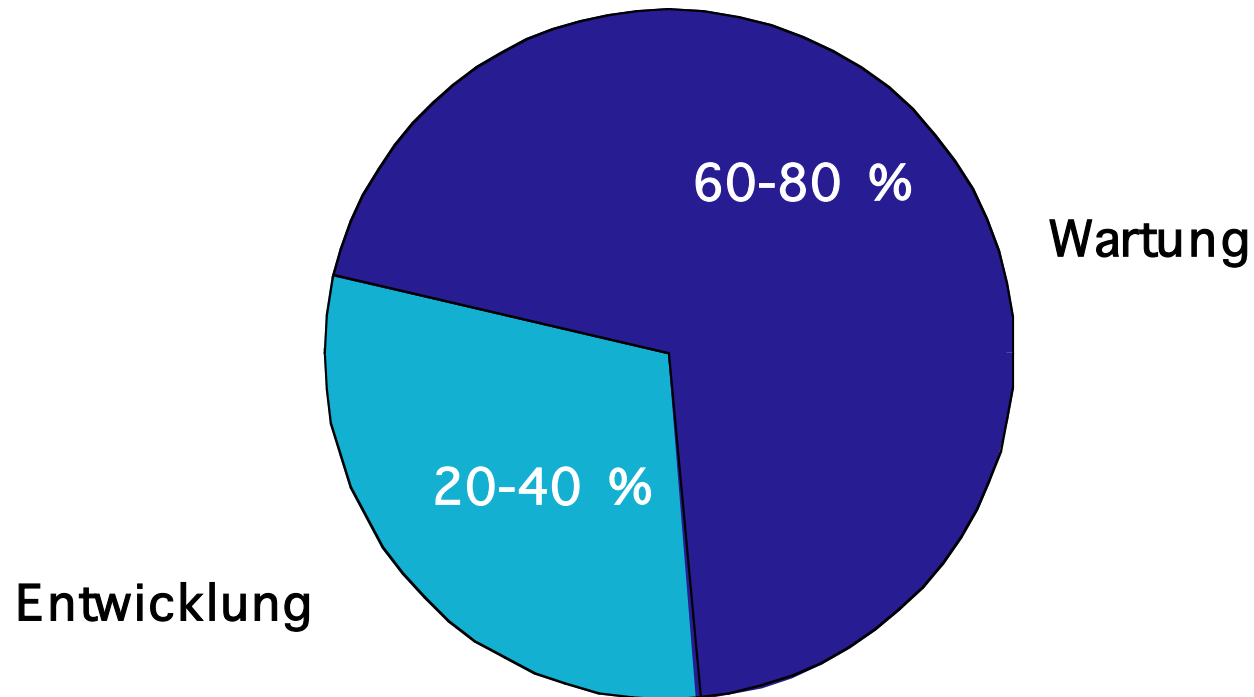


Quelle: Hussmann

- Einzelprojekt:
 - Individuelle Anfertigung
 - Schwerpunkt: Pflege (Fehlerbehebung)

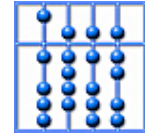


Wartungsaufwand langlebiges Projekt

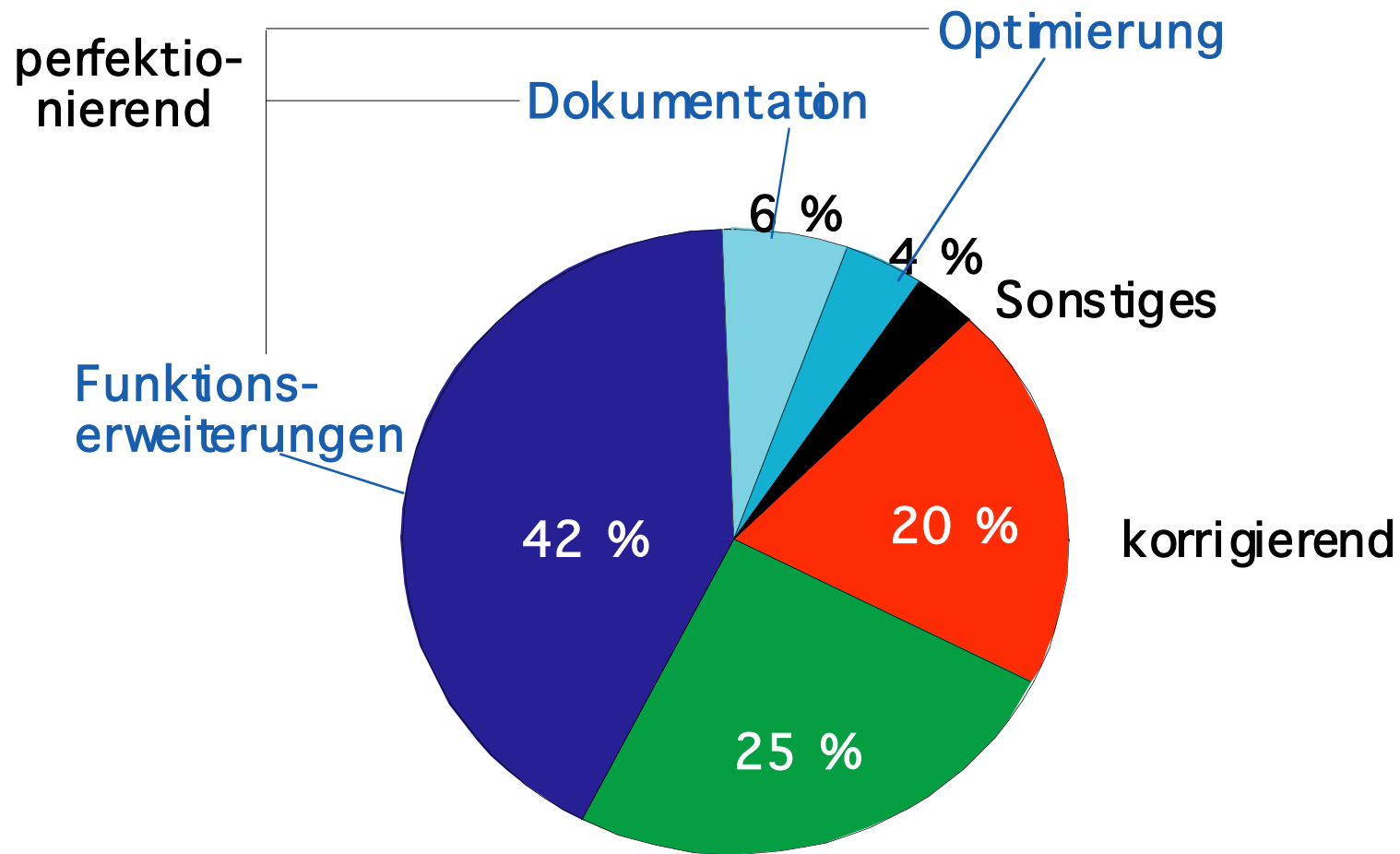


Quelle: Hussmann

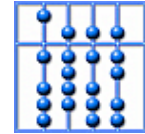
- Langlebiges Projekt:
 - Wiederholte Installation bzw. (anpassbare) Standardanwendung
 - Schwerpunkt: Änderung (Funktionserweiterung)



Aufschlüsselung Wartungsaufwände

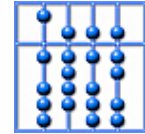


Quelle: Hussmann



Konfigurationsmanagement

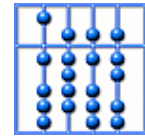
- Ziel:
 - Produkt ist hinsichtlich seiner funktionellen (Code) und äußeren (zugehörige Information) Merkmale jederzeit identifizierbar
- Funktion:
 - Sicherstellung der Identifizierbarkeit, Verfolgbarkeit, Kontrollierbarkeit, Status
 - Darstellung der Zusammenhänge und Unterschiede zwischen verschiedenen Konfigurationen
 - Sicherstellung der Verfügbarkeit früherer Konfigurationen
 - Sicherstellen der Integrität (Gültigkeit, Reifegrad) von Produkten
- Ansatz: Explizite Verwaltung von Produkten



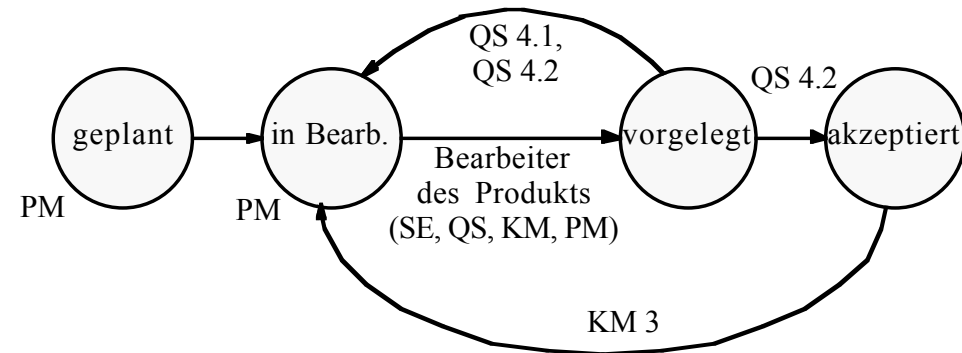
Produkt und Aktivität

von		Produkt	nach	
Aktivität	Zustand		Aktivität	Zustand
SE 1	akzeptiert	Anwenderforderungen	—	—
SE 2.1	in Bearb.	Systemarchitektur	—	—
—	—	Systemarchitektur. <i>Anforderungszuordnung</i>	SE 1, SE 2.5, SE 2.6, SE 3, SE 4- SW, PM 4, PM 5	vorgelegt

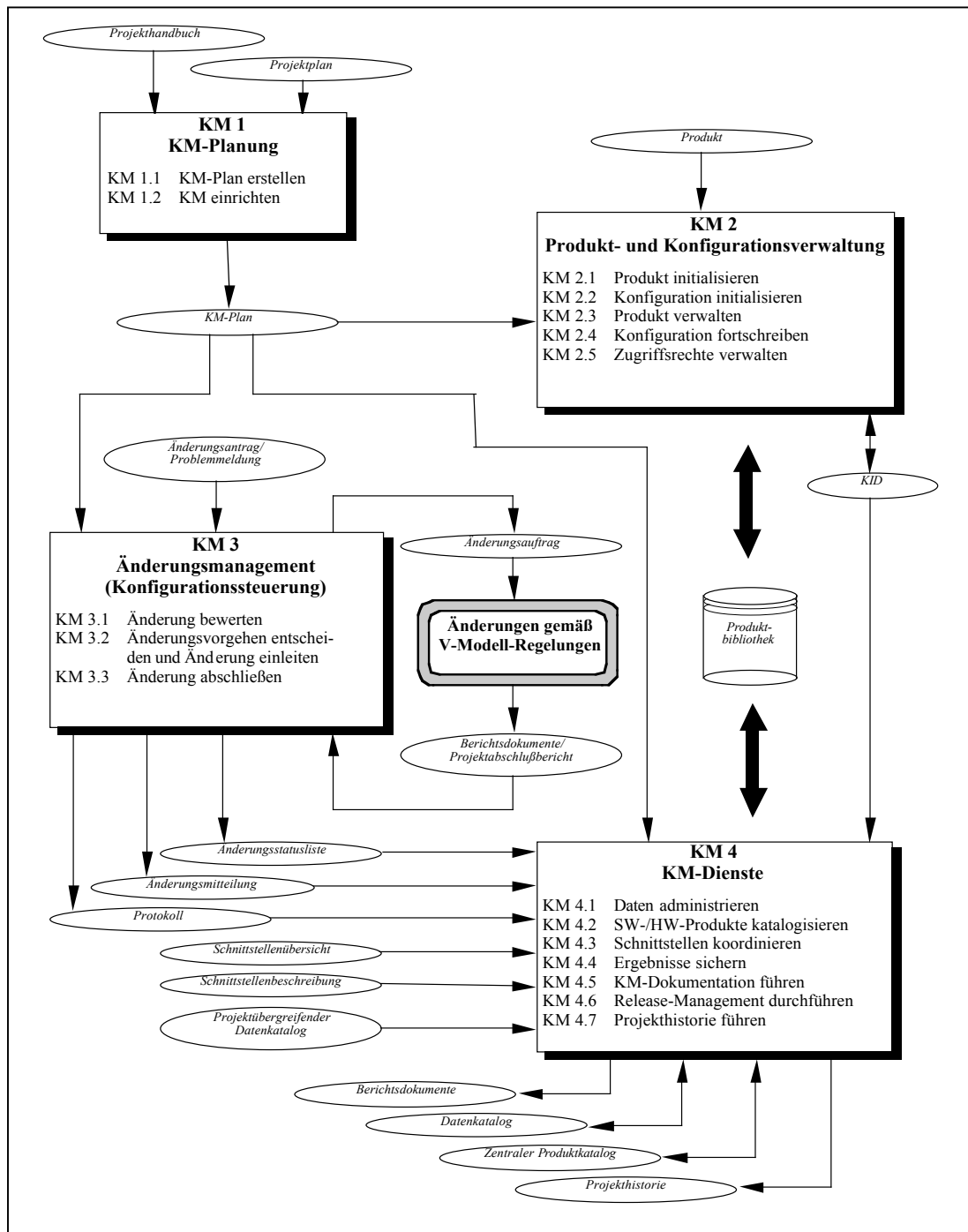
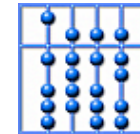
- Aktivitäten:
 - Benötigen, erzeugen, modifizieren Produkte
- Produkte:
 - Synchronisieren, beeinflussen Aktivitäten



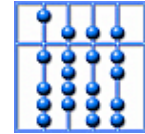
KM: Produktzustände



- Produktzustände:
 - Definiert auf allen Produkten des V-Modells
 - Verbinden KM mit allen weiteren Modulen
 - Ermöglichen Produktverwaltung
- Zusammenhänge:
 - KM: Erfassung Produkt, Produktstatus, Produktänderung
 - QS. Überprüfung/Abnahme Produkt
 - PM: Planung Produkt

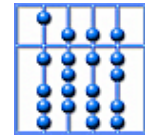


- Planung/Initiierung
- Produkt/Konfig-Verwaltung:
 - Initialisieren Produkte
 - Zustandsverfolgung von Produkten
 - Konfig-Verwaltung
- Änderungsmanagement:
 - Änderungswünsche verwalten
 - Verantwortlichkeiten zu Rollen



Aufgaben Konfigurationsmanagement

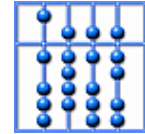
- Planung Konfigurationsmanagement
- Bereitstellen Produkt-/Konfigurationsmanagement
- Bereitstellung Änderungsmanagement
- Bereitstellung von elementaren Diensten
 - Daten administrieren:
 - Zentrale, projektübergreifende Haltung aller Daten
 - Konsistente, vereinheitlichte Datendefinitionen
 - SW-/HW-Produkte katalogisieren: Vorbereitung Wiederverwendung
 - Schnittstellen koordinieren: Sicherung kompatibler Schnittstellen
 - Ergebnisse sichern: Wahrung des erreichten Projektstands
 - KM-Dokumentation führen zur Erstellung von Detailunterlagen und Übersichten verschiedenster KM-Belange,
 - Release-Management: Kontrollierte Konfigurationsfreigabe- und verteilung
 - Projekthistorie führen: Nachvollziehbare Dokumentation über den Projektverlauf



Konfiguration

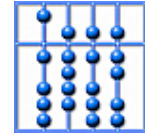
- Konfiguration:
 - Definition: Ansammlung von Produkten, die unabhängig von einander variiert werden können, einschließlich ihrer Beziehungen

- Beschreibung: Konfigurationsidentifikationsdokument
 - Funktion: Erfassung der Konfigurationselemente und ihrer Beziehungen
 - Umfasst:
 - Unterkonfiguration (z.B. HW-Konfiguration, SW-Konfiguration)
 - Konfigurationselemente
 - Hilfselemente (Generierungswerkzeuge)



Konfigurationselement

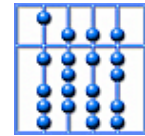
- Konfigurationselement (engl. Configuration Item):
 - Definition: vom Konfigurationsmanagement erfasst und als Einheit behandelte Ansammlung von Hard- und Software
 - Beispiele:
 - Quellcode-Dateien
 - Testtreiber, Testfälle, Testprotokolle
 - Entwicklungsdokumente (z.B. Pflichtenheft, Architekturentwurf)
 - Hardwarekonfigurationsbeschreibung
 - Im allgemeinen:
 - Nur ursprüngliche Dokumente
 - Keine generierten Dokumente (z.B. Binärcode)



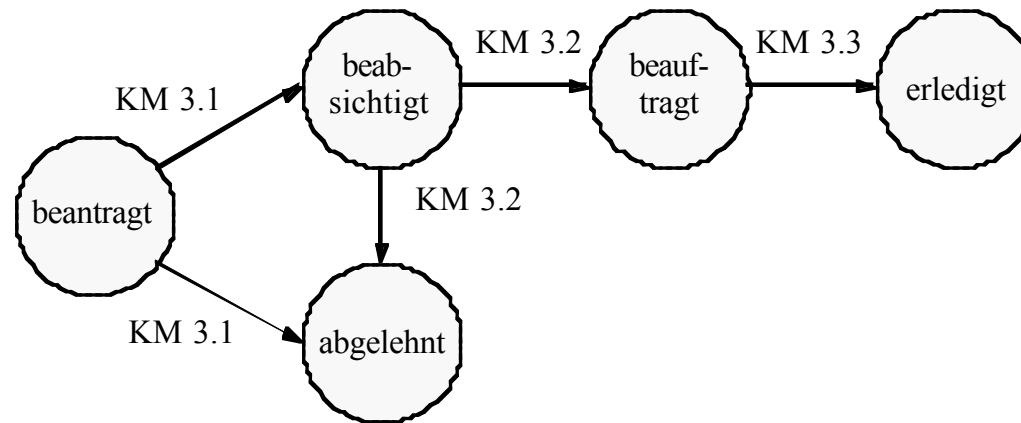
Änderungsmanagement

- Ziel: Kontrolle der Änderungen
 - zur Sicherung der Produktqualität
 - Änderungsumfang
 - Verantwortlichkeiten
 - zur Kontrolle der Projektdurchführung
 - Statusverfolgung
 - Ressourcenplanung

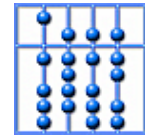
- Aufgaben:
 - Änderungen identifizieren
 - Änderungen bewerten
 - Änderungsvorgehen entscheiden und Änderungen einleiten
 - Änderungen abschließen



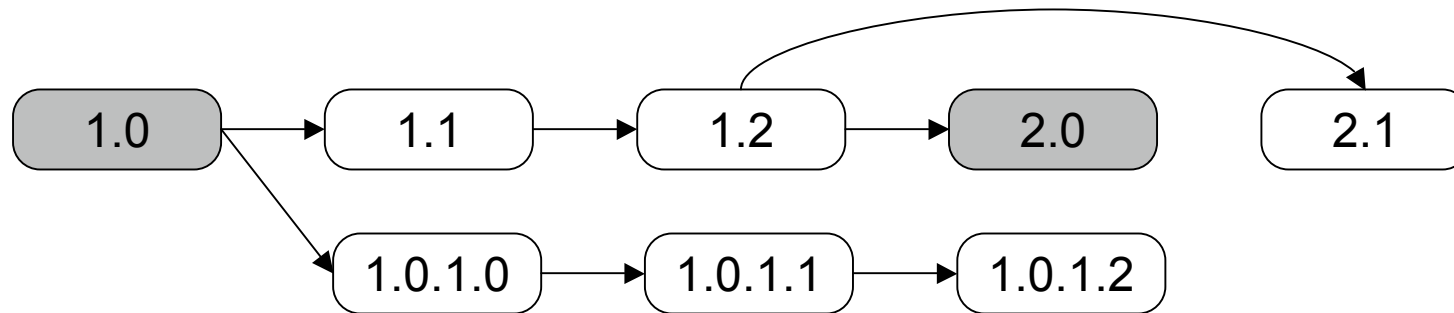
Änderungsmanagement



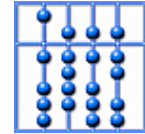
- Änderungen:
 - Änderungen verwalten
 - Entgegennehmen/bewerten
 - Entscheiden/einleiten
 - Abschließen
 - Verantwortlichkeiten für Änderungsaufgaben zuweisen
 - Entscheidungen herbeiführen und dokumentieren
 - Projekthistorie mitführen



Begriffe: Version, Variante, Release

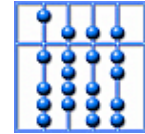


- Begriffe:
 - Version: Ausprägung eines Konfigurationselements zu einem bestimmten Zeitpunkt
 - Variante: Zeitgleich nebeneinander liegende Ausprägung von Elementen
 - Parallele Entwicklungslinien
 - Unterschiedliche Implementierungen derselben Schnittstelle
 - Unterstützung unterschiedlicher HW/SW-Plattformen
 - Release: Element einer Referenzkonfiguration (baseline)
 - Baseline: Zu einem bestimmten Zeitpunkt ausgewähltes, gesichertes und freigegebenes (Zwischen-)Ergebnis



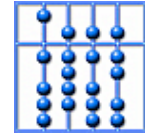
Schnittstelle Berichtswesen

- Berichtswesen
 - Festlegung Verantwortlichkeiten
 - Verteilen Informationen
 - Dokumentation Entscheidungen
- Wesentliche Schnittstellen:
 - Änderungsmanagement: Erfassung Fehler, Änderungen
 - Projekthistorie: Dokumentation Projektabwicklung
- Berichtsarten (Beispiele):
 - Fehleridentifikation
 - Änderungsanträge und Entscheidungen
 - Änderungsmeldungen
 - Projekthistorie



Fehleridentifikation

- Assoziierte Informationen:
 - Fehlernummer (laufende Nummer)
 - Datum des Erfassens
 - Art des Tests (Review/Walkthrough, Test)
 - Evtl. Testdaten
 - Beschreibung des Fehler/Verletzte Anforderung/Zugehöriger Testfall
 - Kritikalität des Fehlers
 - Lokalisierung/Modul/Dokument
 - Version des betroffenen Moduls/Dokuments
 - Aufwand des Findens
 - Maßnahme (Änderung Anforderungen/Änderung Dokument/Keine Änderung)
 - Fehlerstatus: Gefunden/Behoben/Getestet
 - Datum Behebung
 - Aufwand Fehlerbehebung
 - Von Behebung betroffene Module/Dokumente



Projekthistorie

- Entwicklungsverlauf
 - Entscheidungen
 - Schwierigkeiten, deren Ursachen und Behebung
- Änderungs-/Versionshistorie
 - Nachvollziehbarkeit von Änderungen (Änderungen-Version-Bezug)
 - Dokumentation der Änderungshistorie (Anträge und Änderungen)
 - Dokumentation der Änderungsentscheidungen
- Planungsstatistik
 - Planabweichungen und Planänderungen
 - Analyse der Ursachen
 - Möglichkeiten zur Verhinderung weiterer Planabweichungen
- Änderungs- und Fehlerstatistik
 - Änderungsantrag/Problemmeldung
 - Betroffene Produkte
 - Klassifizierung, Beurteilung, Diagnose/Ursache
 - Maßnahmen zur zukünftigen Vermeidung
- Analyse/Auswertung der Projekthistorie („lessons learned“)