

Begriffsdefinitionen

Konfigurationsmanagement

KM stellt sicher, dass Produkte eindeutig identifizierbar sind, Zusammenhänge und Unterschiede von verschiedenen Versionen einer Konfiguration erkennbar bleiben und Produktänderungen nur kontrolliert durchgeführt werden können. Arten der Durchführung: - Automatisch (z.B. Git) - Manuell - Semi-automatisch (z.B. unterstützende Plugins)

**Build-Management**  
Unter Build-Management wird diejenige Funktion verstanden, die alle Bauteile einer Konfiguration erzeugt, die nicht direkt vom Benutzer erstellt oder durch das System vorgegeben wurden. Zentrale Fragen: Wie ist die Software zusammengestellt? Welche Abhängigkeiten müssen berücksichtigt werden? Beispiele für klassische Abhängigkeiten: o Entwicklungsumgebung: Editor, Generatoren, Compiler, Linker, ... o Basissoftware: LAN, ... o Betriebssystem o Hardware

**Change-Management**  
Im Änderungswesen muss definiert werden, wie Änderungswünsche (Change Requests) erfasst werden, wie und durch wen diese bewertet werden und wie die Durchführung von Änderungen zu erfolgen hat. - Versions-Management (Wer hat wann was geändert?) Versionsmanagement befasst sich (in erster Linie mit der Verwaltung der zeitlich aufeinander folgenden Revisionen eines Dokuments.) **Release-Management**  
Ein Release ist eine an Kunden ausgelieferte Konfiguration eines (Software-)Systems, bestehend aus ausführbaren Programmen, Bibliotheken, Dokumentation, Quelltexten, Installationsskripten und so weiter. Das Release-Management dokumentiert ausgelieferte Konfigurationen und stellt deren Rekonstruierbarkeit sicher. Geplant

Produktzustände

**In Bearbeitung**  
Das Produkt wird bearbeitet. Es befindet sich entweder im privaten Entwicklungsbereich des Entwicklers oder unter Kontrolle des Entwicklers innerhalb der Produktbibliothek. **Vorgelegt**  
Das Produkt ist aus der Sicht des Erstellers fertig und wird der Konfigurationsverwaltung übergeben. Ab jetzt kann es einer Prüfung durch die Qualitätssicherung unterzogen werden. Wird das Produkt hierbei abgelehnt, so geht es wieder in den Zustand in Bearb. zurück, andernfalls rückt es in den Zustand akzeptiert vor. Ab dem Zustand vorgelegt an kann der Ersteller nur unter Fortschreibung der Versionsangabe Modifikationen durchführen. **Akzeptiert**  
Das Produkt wurde durch die QS überprüft und freigegeben. Es darf nur noch innerhalb einer neuen Version geändert werden.

**CTL**  
**Pfadoperatoren:**  
-  $A\phi$  - auf allen Pfaden folgt  $\phi$  (englisch: All)  
-  $E\phi$  - auf mindestens einem Pfad folgt  $\phi$  (englisch: Exists)

**Pfad-spezifische Operatoren:**  
-  $X\phi$  - unmittelbar folgt  $\phi$  (englisch: neXt state)  
-  $F\phi$  - irgendwann folgt  $\phi$  (englisch: some Future state oder Finally)  
-  $G\phi$  - auf dem folgenden Pfad folgt in jedem Zustand  $\phi$  (englisch: Globally)  
-  $\phi U \psi$  -  $\phi$  folgt bis zum Erreichen des Zustands  $\psi$  (englisch: Until)  
-  $\phi W \psi$  -  $\phi$  folgt immer oder bis zum Erreichen des Zustands  $\psi$  (englisch: Weak Until)  
-  $EX\phi$  - in (mind.) einem nächsten Zustand gilt  $\phi$   
-  $EF\phi$  - in (mind.) einem der folgenden Zustände gilt  $\phi$   
-  $EG\phi$  - es gibt (mind.) einen Pfad, so dass  $\phi$  entlang des ganzen Pfades gilt  
-  $E[\phi U \psi]$  - es gibt einen Pfad, für den gilt: bis zum ersten Auftreten von  $\psi$  gilt  $\phi$   
-  $AX\phi$  - in jedem nächsten Zustand gilt  $\phi$   
-  $AF\phi$  - man erreicht immer einen Zustand, in dem  $\phi$  gilt  
-  $AG\phi$  - auf allen Pfaden gilt in jedem Zustand  $\phi$   
-  $A[\phi U \psi]$  - es gilt immer  $\phi$  bis zum ersten Auftreten von  $\psi$

**Semantik**  
-  $T(s_0) \models \neg \phi \iff T(s_0) \not\models \phi$   
-  $T(s_0) \models \phi \vee \psi \iff T(s_0) \models \phi$  oder  $T(s_0) \models \psi$   
-  $T(s_0) \models EX\phi \iff T(s_1) \models \phi$   
-  $T(s_0) \models EG\phi \iff \forall i : T(s_i) \models \phi$   
-  $T(s_0) \models \phi EU \psi \iff \exists k : T(s_k) \models \psi \wedge \forall i < k : T(s_i) \models \phi$

**Transformationen**  
-  $\neg A\phi \equiv E\neg\phi$   
-  $\neg AF\phi \equiv EG\neg\phi$   
-  $\neg EF\phi \equiv AG\neg\phi$   
-  $\neg AX\phi \equiv EX\neg\phi$   
-  $AG\phi \equiv \phi \wedge AXAG\phi$   
-  $EG\phi \equiv \phi \wedge EXEG\phi$   
-  $AF\phi \equiv \phi \vee AXAF\phi$   
-  $EF\phi \equiv \phi \vee EXEF\phi$   
-  $A[\phi U \psi] \equiv \psi \vee (\phi \wedge AXA[\phi U \psi])$   
-  $E[\phi U \psi] \equiv \psi \vee (\phi \wedge EXE[\phi U \psi])$

Wahrheitstabelle				
$p$	$q$	$p \wedge q$	$p \vee q$	$p \rightarrow q$
$T$	$T$	$T$	$T$	$T$
$T$	$F$	$F$	$T$	$F$
$F$	$T$	$F$	$T$	$T$
$F$	$F$	$F$	$F$	$T$