

Systematische Unterstützung des Entwicklungs- und Versionierungsprozesses

Thomas Schröders
01.10.2009

Agenda

- Identifikation von Aufgabengebieten im Softwareentwicklungsprozess
- Was ist für die Teamarbeit wichtig?
- Beispiel für einen dynamischen Entwicklungsprozess

Softwareentwicklungsprozess vs. Vorgehensmodell

Wikipedia: (http://en.wikipedia.org/wiki/Software_development_process)

A software development process is a structure imposed on the development of a software product. (...)

There are several models for such processes, each describing approaches to a variety of tasks or activities that take place during the process.

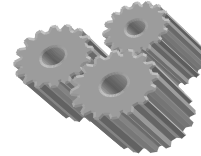
Aufgabengebiete



Aufgabengebiete

- Blickwinkel des Teams

Was leistet XYZ (für das Team)?



Was benötigt XYZ (vom Team) dazu?

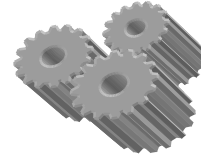
Aufgabengebiet: WAS soll wie gemacht werden?

■ Requirements Engineering

Interessen und Bedürfnisse erarbeiten
Analysemodell liefern

- Anforderungen gliedern und transparent darstellen
- Abhängigkeiten identifizieren und verwalten

Änderungs- und Versionierungsprozess gestalten



Ansprechpartner und Zeit
Rückmeldung vom Team über

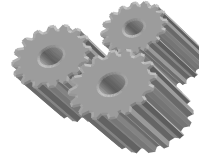
- getroffene Entscheidungen
- Realisierungsstand

guter Überblick, zentraler Datenhaushalt

Aufgabengebiete: Was soll **WIE** gemacht werden?

- Software Engineering

Architektur und Design definieren
System modellieren
Implementierungseinheiten bilden, verwalten und realisieren
Änderungen protokollieren und archivieren
Änderungs- und Versionierungsprozess gestalten

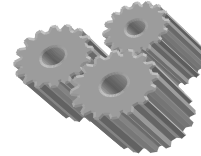


Entwicklungsumgebung zur Modellierung und Implementierung
Managementumgebung für Implementierungseinheiten
- Transparenz über Anforderungen und Priorisierungen
- Änderungsmanagement und Versionskontrolle
Automatisierung wiederkehrender Aufgabenbestandteile

Aufgabengebiet: Supportfunktionen

■ Releasemanagement

Produkt liefern
Requirements Engineering und Software Engineering vereinigen
Releaseplan abstimmen
Branch- und Mergeprozesse steuern
Installationsprozess definieren und Umgebungen aufbauen
Gelieferte Produkte versionieren und archivieren

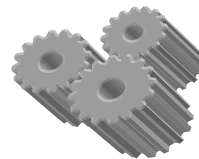


guter Überblick über
- Realisierungsstand
- Abhängigkeiten zwischen Implementierungseinheiten
- Abhängigkeiten zu Anforderungen
zentraler oder gut integrierter Datenhaushalt
Automatisierung wiederkehrender Aufgabenbestandteile

Aufgabengebiet: Supportfunktionen

■ Qualitätsmanagement

Sichtweise auf bzw. Vorstellung vom Produkt homogenisieren
Nutzen der Arbeitsergebnisse für andere sicherstellen
Testaktivitäten steuern und Tests unterstützen
Automatisierte Tests überwachen
Richtlinien und Metriken definieren und deren Einhaltung prüfen
Qualitätsstatus berichten



Ansprechpartner und Zeit
transparenter Überblick über

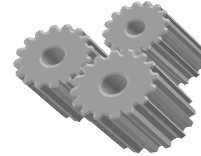
- Anforderungen
- Realisierungsstand
- Releaseplanung

Testumgebung für Akzeptanz- und Integrationstests

Aufgabengebiet: Supportfunktionen

■ Projektmanagement

Einhaltung von Projektzielen, Budget und Zeitplan sicherstellen
Rahmenbedingungen schaffen
Kommunikation und Informationsfluss sicherstellen
Projektstatus berichten
Änderungsprozess gestalten



Ansprechpartner und deren Zeit
Möglichkeit zur flexiblen Projektplanung
Überblick über den Projektfortschritt
- Realisierungsstand
- Releaseplanung
- Budget

Agenda

- Identifikation von Aufgabengebieten im Softwareentwicklungsprozess
- Was ist für die Teamarbeit wichtig?
- Beispiel für einen dynamischen Entwicklungsprozess

Analyse 1: Leistungen für das Team

- Gemeinsamkeiten:

- Aufgaben verwalten

Änderungsprozess

- Aufgaben verknüpfen

Transparenz

- Aufgaben erledigen

Nutzen für andere

Transparenz über Aufgaben und Abhängigkeiten schaffen
und dadurch einen Änderungsprozess ermöglichen!

Analyse 1: Leistungen für das Team, Beispiel

- Requirements Engineering identifiziert folgende Geschäftsvorfälle:
 - Konto eröffnen
 - Kontodaten ändern
 - Konto auflösen

- Software Engineering plant daraus folgende Implementierungseinheiten:
 - Kontomanagement basic
 - Kontomanagement advanced
 - Kontomanagement extended

Analyse 2: Forderungen an das Team

- Gemeinsamkeiten:

- Kommunikationsbedarf

Planung / Status

- Transparenter Überblick

Abhängigkeiten

- Automatisierung

Ballastfreies Arbeiten

Konzentration auf das Wesentliche durch Automatisierung
und Nutzung der Abhängigkeiten (Navigation).

Analyse 2: Forderungen an das Team, Beispiel

- Definition einer einheitlichen Sprache
- Nutzung der definierten Bezeichnungen für Entitäten, Prozesse und Eigenschaften
- Wiederholung der Beschreibung durch Navigierbarkeit vermeiden
- Keine blockierenden Tätigkeiten, volle Konzentration auf wesentliche Aufgaben

Agenda

- Identifikation von Aufgabengebieten im Softwareentwicklungsprozess
- Was ist für die Teamarbeit wichtig?
- Beispiel für einen dynamischen Entwicklungsprozess

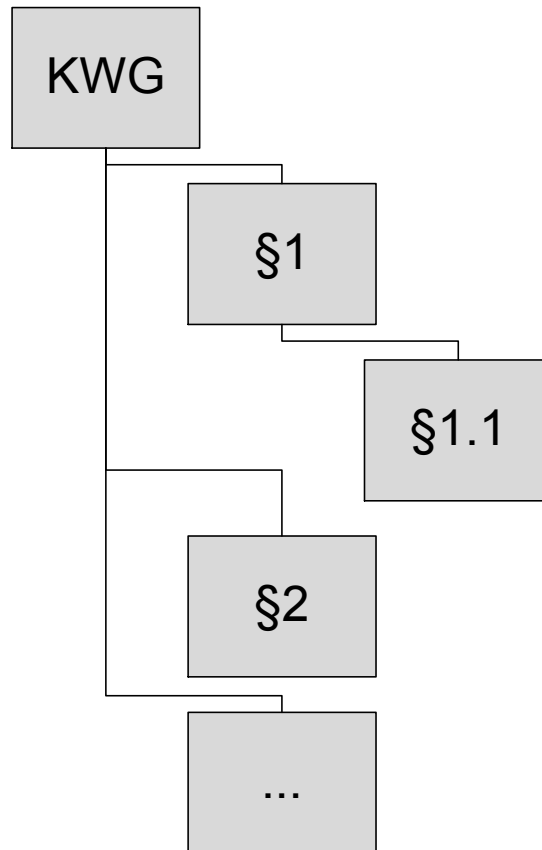
Praxisbeispiel: Nutzung von Verlinkung

- Universeller Einsatz von Mantis* für die Verwaltung von
 - Rahmenbedingungen
 - Analysemodellen
 - Implementierungseinheiten
 - Releases
 - und insbesondere Abhängigkeiten
- Verknüpfung von Aufgaben über mehrere Arbeitsgebiete hinweg

* <http://www.mantisbt.org/>, web-based bugtracking system (PHP, MySql) unter GNU GPL

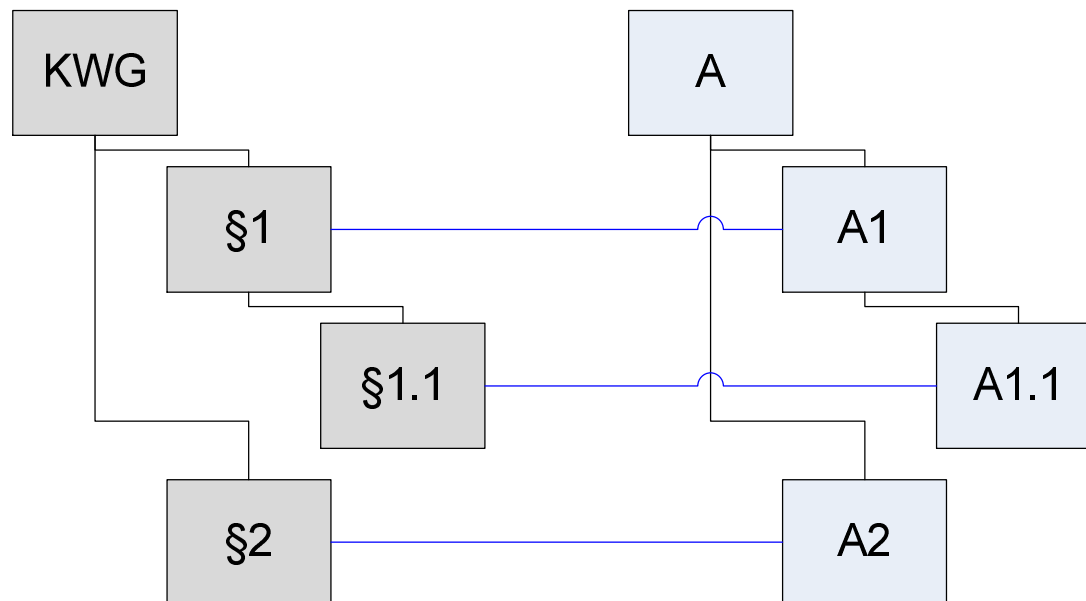
Praxisbeispiel: Nutzung von Verlinkung

- Zentrale Erfassung von Vorgaben



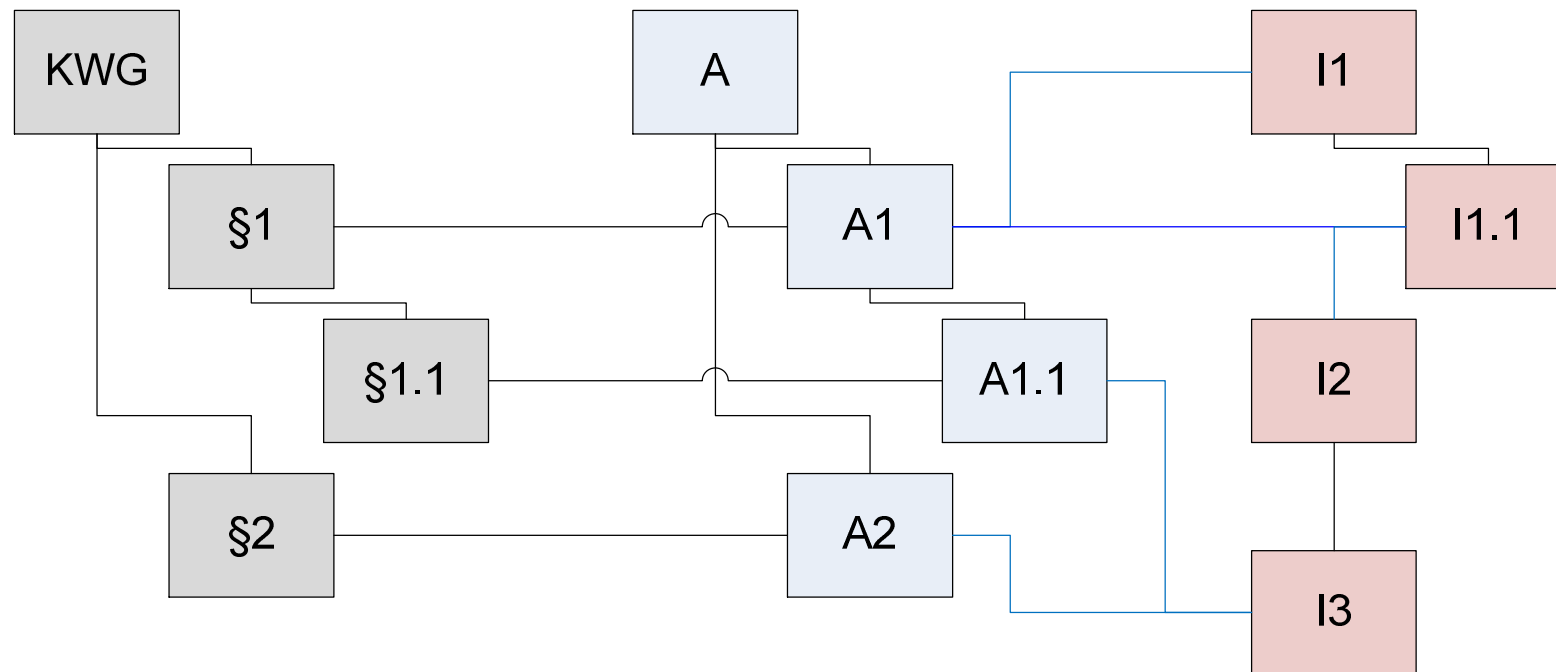
Praxisbeispiel: Nutzung von Verlinkung

- Erstellung eines Analysemodells
- Verknüpfung des Analysemodells mit den Vorgaben



Praxisbeispiel: Nutzung von Verlinkung

- Ableitung von Implementierungseinheiten
- Verknüpfung der Implementierungseinheiten mit den Anforderungen – und damit auch mit den Vorgaben



Praxisbeispiel: Nutzung von Verlinkung

- Zusammenfassung
 - Zentrale Erfassung der Rahmenbedingungen
 - Verknüpfung der Aspekte des Analysemodells mit den Rahmenbedingungen
 - Verknüpfung der Implementierungseinheiten mit den Aspekten des Analysemodells
 - Verknüpfung des Releases mit den Implementierungseinheiten

Danke für die Aufmerksamkeit!

- Fragen und Antworten

