Seminar Sicherheit und Softwareengineering

Sommersemester 2013

LS14
Arbeitsgruppe
Software Engineering for Critical Systems

11.4.13

Outline

Vorstellung der AG

Das Seminar - Wichtige Meta-Fähigkeiten

Werbung

Abschlussarbeiten

- Themen siehe: `http://www-jj.cs.tu-dortmund.de/secse/pages/teaching/thesis/index_de.shtml`
- Seminarthemen bereiten Abschlussarbeitsthemen vor

Hilfskräfte

- Themen siehe: `http://www-jj.cs.tu-dortmund.de/secse/pages/home/jobs_de.shtml`
- Mitarbeit in verschiedenen Projekten

Ablauf

Leistungsbestandteile

- Kommentierte Gliederung
- Review-Fassung
- Reviews
- Abgabe Ausarbeitung
- Abgabe Folien
- Vortrag
- Diskussion

Betreuung

- Vorgespräch (Verständnisfragen)
- Besprechung der Gliederung
- Besprechung der Reviews/ der Reviewfassung
- Besprechung der Folien

Ausarbeitung

Umfang

- ca. 15 Seiten Hauptinhalt, nicht mit gerechnet:
  - Titelblatt
  - Inhalts- / Tabellen- / Abbildungsverzeichnis
  - Bibliographie
- min 10 Seiten Reintext
  - Ohne Abbildungen
  - Ohne Kapitelumbrüche

Vorlagen (Bitte Einhalten)

Liegen im Latex und Word Format vor

`http://www-jj.cs.tu-dortmund.de/secse/pages/teaching/allgemeineInfo/index_de.shtml`

Ausarbeitung II

Inhalt

- Verständliche Darstellung des Inhalt
  - Zielgruppe: Studenten mit abgeschlossenem Bachelor
  - Selfcontainment: Erklären der benötigten Begriffe
- Fazit mit eigener kritischer Stellungnahme

Einstiegsliteratur

Wichtig: Nutzung weitergehender Literatur!

Gliederung

- Kapitelüberschriften
- Kurze Übersicht über die Kapitelinhalte (ca. 100 Worte pro Kapitel)
- Literaturübersicht

Review

Zwei Reviews

- Jeder bekommt zwei Reviews
- Jeder erstellt zwei Reviews

Inhalt und Form

- ca. 1 Seite
- Kurze Zusammenfassung
- Positive Punkte
- Problem Punkte
- Verbesserungsvorschläge

Vortrag

Umfang

- Vortragsdauer: 35 Min (30-40 Min ok)
- anschließend Diskussion

Beamer und Präsentationsrechner (PDF) stehen zu Verfügung.

Zum Inhalt

- Spannungsrahmen erzeugen
- Benötigte Grundlagen kurz aber ausreichend

Was selbstverständlich sein sollte....

Plagiat

Durchgefallen und Benachrichtigung des Prüfungsausschusses!

Verspätete Abgabe

- Ohne Absprache wird die Teilleistung mit 5 bewertet
- Absprache muss von Betreuer bestätigt werden

Anwesenheit

Bei allen Vorträgen ist die Anwesenheit Pflicht!

Abgabeformat

PDF + eine gedruckte Fassung

Zeitplan

11.04.13 (14:00)	Themenvorstellung
14.04.13 (24:00)	Rückmeldung
06.05.13 (24:00)	Abgabe Gliederung
10.06.13 (24:00)	Abgabe Vorversion Ausarbeitung
01.07.13 (24:00)	Abgabe Reviews
15.07.13 (24:00)	Abgabe Ausarbeitung
22.07.13 (24:00)	Abgabe Folien
22.-26.7.13	Vorträge

Noten...

Ausarbeitung und Gliederung 40%

Struktur, Verständnis, Form, Inhalt, Quellen, ...

Review 10%

Struktur, “Hilfeleistung“, ...

Vortrag 40%

Verständlichkeit, Aufbau, ...

Teilnahme an der Diskussion 10%

Häufigkeit, Qualität, ...

Themen Rückmeldung

Mail mit 5 Themenwünschen (nach Priorität geordnet)

- ASAP
- vorname.nachname@cs.tu-dortmund.de (Sven Wenzel)
- Name, Matrikelnummer, Studiengang, Semester
- relevante Vorlesungen und Seminare
- weitere qualifizierende Vorkenntnisse

Deadline

Sonntag: 14.04.13 (24:00)

Grundlagen + Anwendung des Modelcheckers VCC

Model Checking

Vergleich des Verhaltens eines Programms mit einer vorgegebenen Spezifikation

- Vielzahl von Tools vorhanden
  - Jeweils für bestimmte Eingabeformate
- Microsoft VCC: Modelchecker für C

Thema

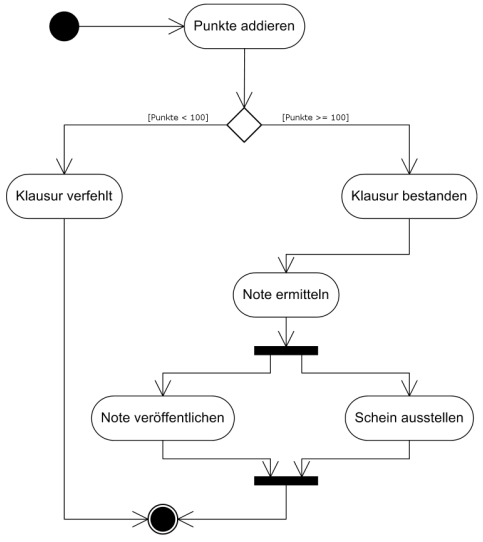
- VCC vorstellen: Grundlagen,
- Funktionsweise + Anwendung erklaren
- Anwendungsbeispiele vorstellen

Analyse von Geschäftsprozessmodellen unter Einbeziehung von Datenmodellierung

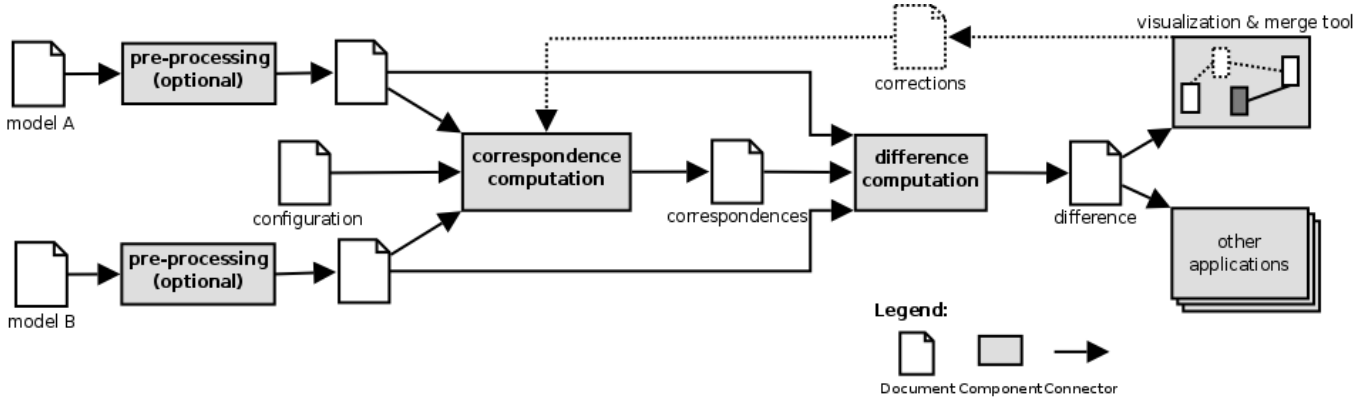
- Es existiert eine Vielzahl von Modellierungssprachen für (Geschäfts-)Prozesse
  - z.B. UML, BPMN, EPK, ...
- Modelle können mit automatischen Verfahren analysiert werden
- $\rightarrow$ Je mehr Informationen im Modell erfasst sind, um so bessere Analysen sind möglich
- Es gibt Modellierungssprachen, die auch explizit betroffene Daten abbilden können
  - z.B. ADEPT bzw. AristaFlow
- Vortrag soll Grundlagen und mögliche Verwendungsbeispiele vorstellen

Grundlagen deklarativer Prozessmodellierung

- Die ”üblichen“ Prozesssprachen bilden explizit die Abläufe zwischen den einzelnen Aktivitäten ab, z.B. UML Aktivitätsdiagramme
- Für manche Situationen umständlich
- $\rightarrow$ Möglicher Ausweg: Deklarative Spezifikation des Modells
  - z.B. DecSerFlow
  - Modellierung und Analysen basieren auf LTL
- Vortrag stellt Grundlagen, Semantik und Analysemöglichkeiten von DecSerFlow vor



Differenzberechnung für Modelle

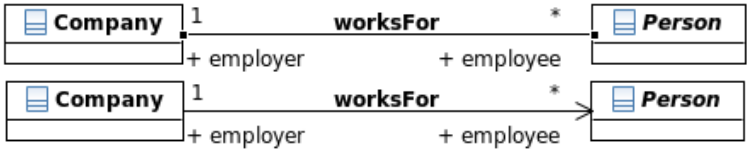


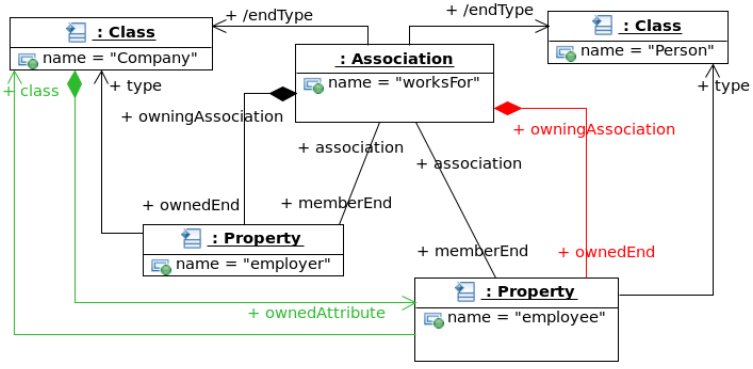
- Modelle in der Softwareentwicklung
- Versionierung
- Was hat sich in einem Modell geändert?
- Matching von Elementen zweier Modelle
- Ableiten der Differenz

- Vortrag stellt Grundlagen und SiDiff-Framework vor

Operationserkennung auf Modelldifferenzen

- aufbauend auf Thema ”Differenzberechnung“
- abgeleitete Differenz of sehr technisch
- Abstraktion auf semantische Ebene

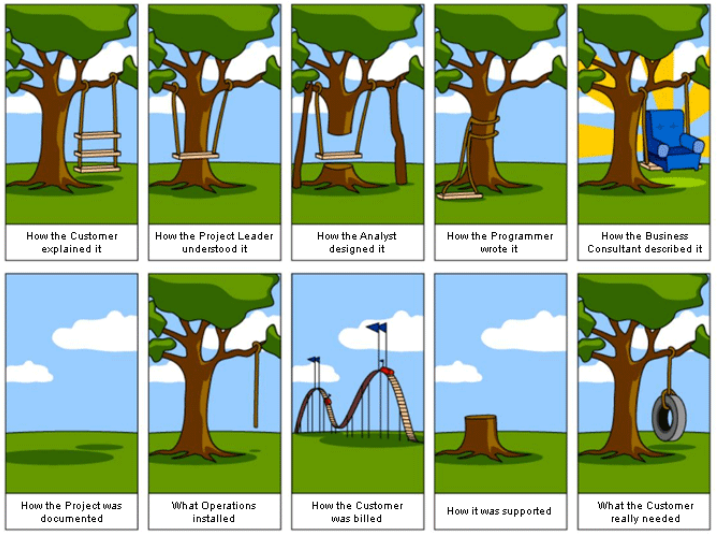




- Vortrag stellt Problem und Lösungsansatz vor
- kurze Einführung in Henshin
- Thema ”Differenzberechnung“ vorausgesetzt (Zweitwunsch)

Verbindliche Spezifikation von Anforderungen

- klare Spezifikation von Anforderungen unumgänglich
- dedizierte Anforderungssprachen



- Vortrag stellt Grundlagen vor und gibt einen Überblick über RSLs
- u.a. soll auch Bezug auf nicht-funktionale Anforderungen (z.B. Sicherheit) genommen werden

Security Patterns

- Design Patterns zur Behandlung von Sicherheitsproblemen
- Kurze Vorstellung vom Pattern-Konzept
  - ein oder zwei Design Patterns zeigen
- Konzept und Beispiele Security Patterns
- Recherche und Vorstellung von Verwendung

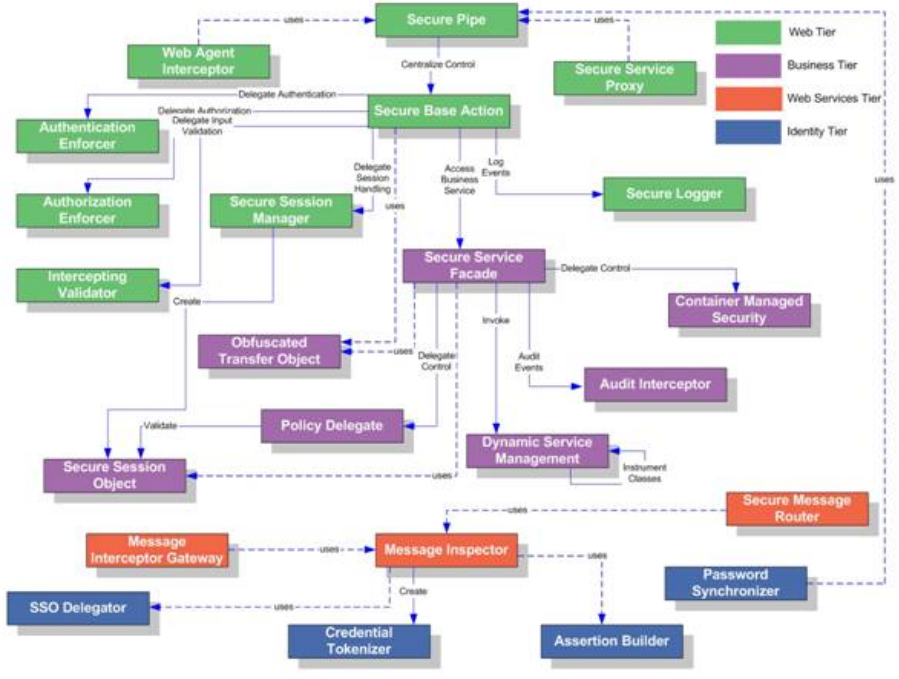
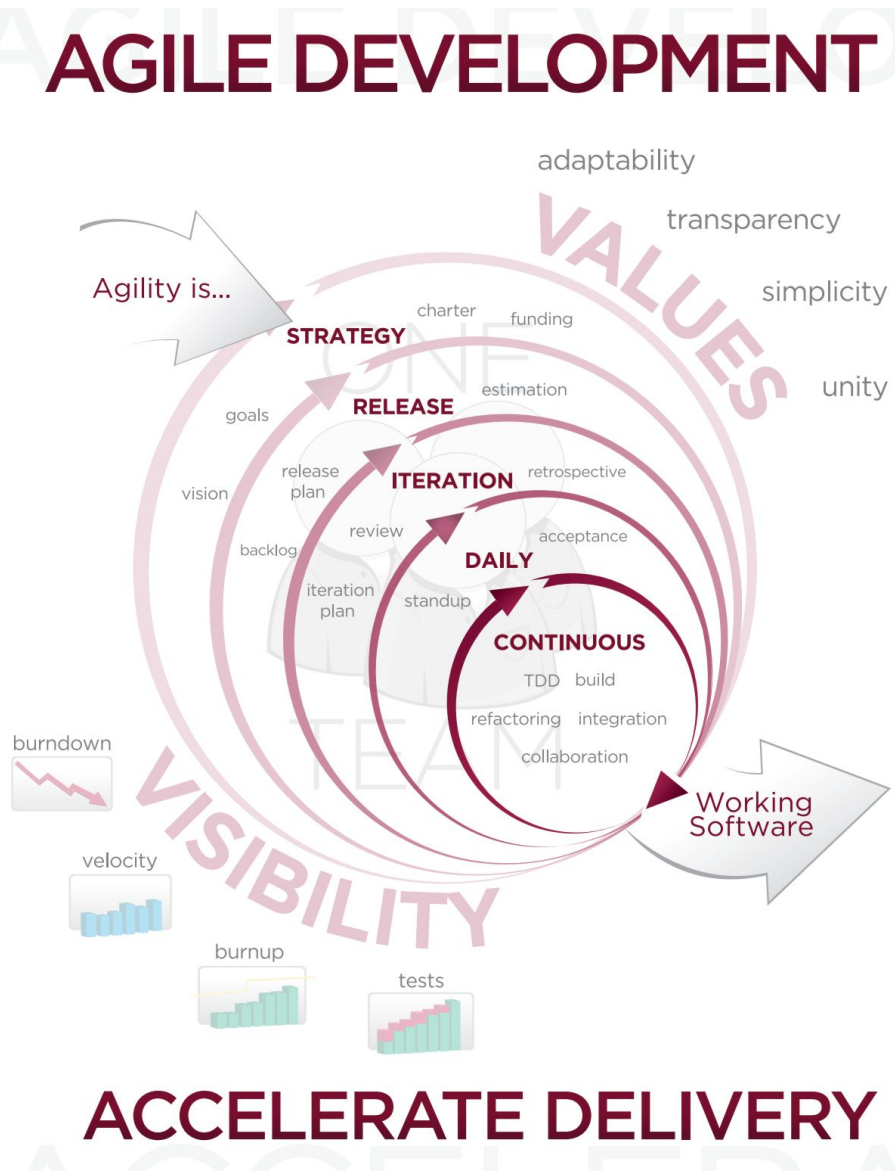


Figure: Quelle: www.coresecuritypatterns.com

Agile Development vs. Security Requirements

- Vorstellung von Agilen Softwaremethoden
- Extreme Programming als Beispiel-Methode, gerne weitere
- Sicherheitsaspekte in agilen Methoden
  - Stand vorstellen
  - Recherche
  - Ansätze aufzeigen



Berechnung von Nash-Gleichgewichten

- Spieltheoretische Sicht:
  - Angreifer sind keine stochastischen Variablen
  - Sondern reagieren auf gewählte Strategie
- Vorstellung eines Algorithmus zur Berechnung der optimalen Lösung (NGG)
- (mit einer wiederholten Polymatrix-Approximation) Baum

Figure: System mit Verteidiger und Angreifer

Baumanalysen (ETA/FTA)

- Recherche und Vorstellung der Verfahren:
- Ereignisbaumanalyse
  - Verfahren zur Bestimmung mgliche Folgen eines auftretenden Fehlers
- Analog: Fehlerbaumanalyse

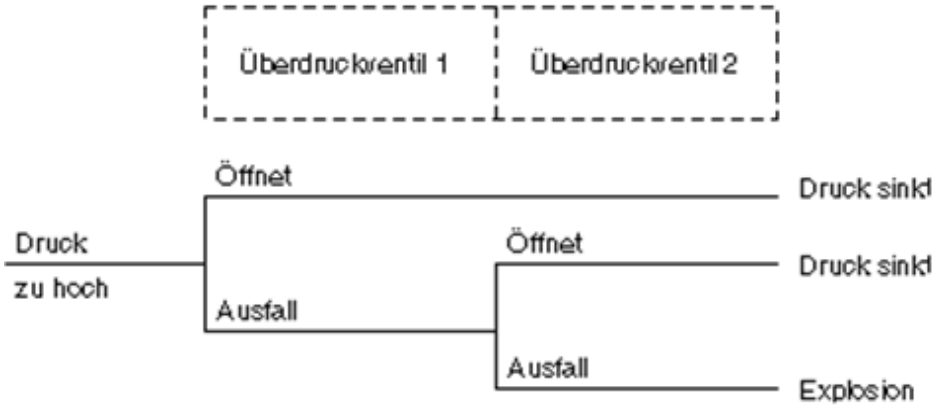
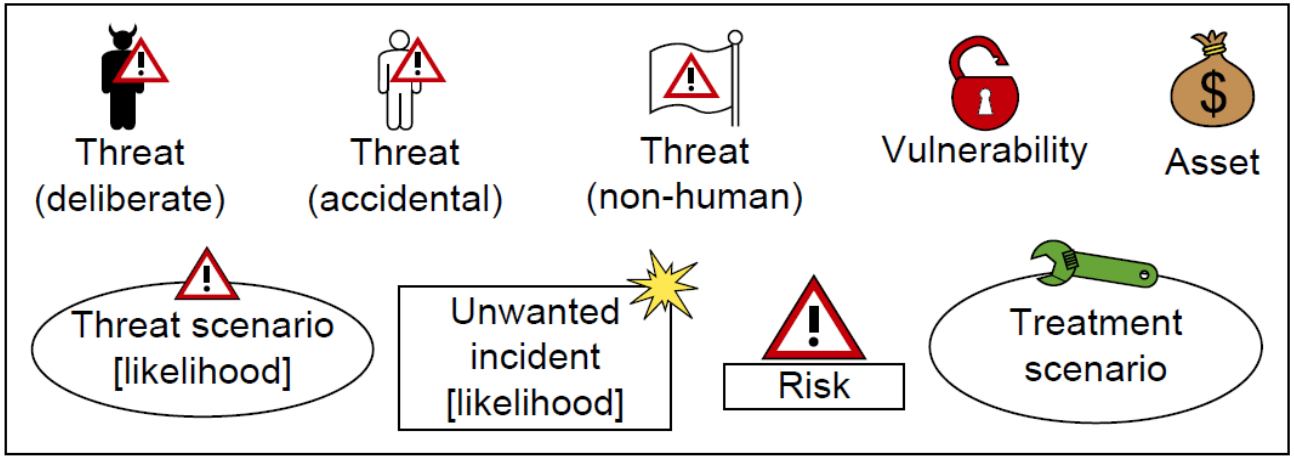


Figure: Beispiel eines Ereignisbaums

CORAS Sicherheitsanalysen



- Erarbeitung und Vorstellung von
- CORAS:
  - Visualisierungssprache
  - Angelehnt an UML-Usecases
  - Darstellung von Sichereitsrisiken
  - Bietet auch eine definierte Methodik

Themen Rückmeldung

Mail mit 5 Themenwünschen (nach Priorität geordnet)

- ASAP
- vorname.nachname@cs.tu-dortmund.de (Sven Wenzel)
- Name, Matrikelnummer, Studiengang, Semester
- relevante Vorlesungen und Seminare
- weitere qualifizierende Vorkenntnisse

Deadline

Sonntag: 14.04.13 (24:00)

Thank you

Questions?