Softwaretechnologie II

Prof. Dr. U. Aßmann
Technische Universität Dresden
Institut für Software- und Multimediatechnik
Gruppe Softwaretechnologie
http://st.inf.tu-dresden.de
WS 11/12-0.1, 10.10.11

Vorlesungen und Übungen

- Vorlesung:
  - Prof. Dr. Uwe Aßmann, Nöthnitzer Str. 46, 2. OG, Raum 2087
  - Katrin Heber, Sekretärin. 0351 463 38 463
  - Sprechstunde Mi, 11:00-13:00. Bitte bei Frau Heber anmelden.
  - Email katrin.heber@tu-dresden.de. Bitte über Frau Heber kontakten, da emails an Prof. Aßmann oft nur verzögert beantwortet werden können

- Vorlesung ist empfohlen für Jahr 3 (Bachelor und Diplom)

- Wichtigste Informationsquelle:
  - http://st.inf.tu-dresden.de/teaching/swt2
  - http://st.inf.tu-dresden.de/ Teaching -> Softwaretechnologie II

- Übungsleiter: Sebastian Richly

  Übungen können nur einen kleinen Teil der Vorlesung abdecken
  Ab Woche 2
  Semester ist in Komplexe aufgeteilt:
  - Ontologien
  - Anforderungsanalyse: ZOPP, Lasten- und Pflichtenheft
  - Testen Regressionstest
  - Reuseware
  - Model Driven Architecture

Übungsgruppen

- Teilung der Übungsgruppen in kleine Gruppen à 4-5 Personen

- Zumeist 2-3 Woche Zeit zur Bearbeitung eines Komplexes

- Lösungen werden ins SVN eingechocket und dann bewertet
  - Nacharbeitung möglich

- Alle Übungskomplexe müssen bearbeitet werden
  - Ansonsten Prüfung nur möglich als 2/0/0 Prüfung

Content

- 01 Introduction
- 02 What is Engineering?
- Part I Analysis and Validation
  - Requirements Analysis
  - Testing
  - Reviews and Inspections
- Part II: Design
  - Functional design
  - Action-oriented design
  - Advanced Object-oriented design
  - Framework-based design
- Part III: Earn Money with Software
  - Part IV: Product Lines
    - Transformational design and MDA
**Recommended Literature: Overview Books**

- **We recommend one of (reading instructions can be followed in one of them):**

- **Other good books, priority from top to bottom:**

**Recommended Books on UML – Unified Modeling Language**

- **UML is required. It is expected that you learn UML yourself from a good book.**

- **We recommend one of:**
  - Online documentation on www.omg.org/uml
  - H. Störrle. UML für Studenten. Addison-Wesley (cheap, good!).

- **Other excellent books:**

**Reference Books**


**Analysis and Design**

- **E. Gamma et. al, Design Patterns, Addison-Wesley, ISBN 0-201-63361-2.**
  - This standard reference book belongs to the bookshelf of every software engineer!
  - Buy this now, if you want to visit "Design Patterns and Frameworks".

- **Others**
  - Rumbaugh et.al. *Object-oriented modelling and design*. Prentice-Hall.
Component-Based Design

- C. Szyperski: Component Software. Addison-Wesley
- K. Czarnecki, U. Eisenecker: Generative Programming. Addison-Wesley

Project Management

- B. W. Boehm, Software Risk Management, 1989
- F. Brooks, The Mythical Man-Month, Addison-Wesley, 1975
- E. Yourdan: The Death March.
- David Thiel. The 12 simple secrets of Microsoft McGraw-Hill.

Implementation


Testing and Quality

- N. Fenton, S.L. Pfeiffer, Software Metrics – a rigorous and practical approach. PWS Publishing.
Configuration Management

- Version control with subversion. http://svnbook.red-bean.com/, also available as paper book of O'Reilly

Miscellaneous Literature

- On Writing:
- Fogler/LeBlanc, Strategies for Creative Problem Solving

Warning: Remarks on the Nature of the Course

- A University is unlike a high school
  - You should not expect to get a book, and that's it
  - Software Engineering is too broad for that, unfortunately
  - The lectures have to focus on most important things
  - You should not expect to be an expert after the course
- Find your way from the lecture slides into the books
  - Follow the reading instructions
  - Learn the additional material and read the additional readings
  - Follow the exercises in the groups
- Expect to learn 3-4 weeks for the oral exam
  - Don't wait until 1 week before the exam! That's too late...
- Be aware: you have not yet seen larger systems
  - Middle-size systems start over 100KLOC

Remarks on the Nature of the Course

- The purpose of lecturing is
  - To give you a condensed insight on the most important topics, such that you do not waste too much time during reading
  - To give you pointers for future work, once you left the course
    - If you haven't got the pointer, you can waste years in darkness
Main Goals

► Learn about “engineering” software
  ■ Engineering attitudes
  ■ Technology, process, experiences, human conditions
► Get as many ideas as possible (broad overview)
  ■ NOT: technical in-depth teaching (this must be left to other courses)
► Get an introduction into the main obstacle: from a set of requirements, how do I arrive at a system? (forward engineering)
► Learn about systematic methods for graph-based specifications
  ■ Because almost all requirements and design notations are graph-based
  ■ Get hold on the complexity of a large specification
► Learn about the behavioral language Petri Nets, and derivatives thereof

Design

► Know different forms of design methods
  ■ functional, object-oriented, data-oriented
► Know behavioral methods to generate code for verifiable specifications
  ■ Petri nets
► Know about “software architecture” and architectural styles

Processes

► Get simple overview of software processes
  ■ MDA, XP, V-model, ...
► Know what inspections are
► Know about maintenance problems
► Know about requirement specification
► Know about basic testing concepts

Earning Money With Software

Business models
Markets
Product lines
Other Courses

► **Design Patterns and Frameworks (WS)**
  - Basic design patterns
  - Design patterns in frameworks
  - Role-based design
  - Composition of design patterns
  - Layered frameworks

► **Component-based Software Engineering (SS)**
  - Black-box component models (e.g., EJB)
  - Grey-box component models (e.g., Aspects)
  - Software composition

► **Software-Management (SWM) (SS)**

► **Software-Werkzeuge (SEW) (WS)**

► **Enterprise Software (ES, Dr. Kubach, SAP, WS)**

---

TU Dresden, Prof. U. Aßmann

Announcements

Folie 21 von 24

---

The End

► **http://st.inf.tu-dresden.de**