

Course "Softwareprozesse"

Introduction and Overview

Lutz Prechelt

Freie Universität Berlin, Institut für Informatik

<http://www.inf.fu-berlin.de/inst/ag-se/>

- Topics of this course
 - 1-slide introduction of each
- Topics of "Softwaretechnik" course
 - and their relationship to those of the present course

Learning objectives

- Understand what topics to expect in the present course
- Understand roughly how they relate to one another
- Review the content of the basic course "Softwaretechnik"
- Understand how the present course complements "Softwaretechnik"

(aus SWT): Taxonomie "Die Welt der Softwaretechnik"

Welt der Problemstellungen:

- Produkt (Komplexitätsprob.)
 - Anforderungen (Problemraum)
 - Entwurf (Lösungsraum)
- Prozess (psycho-soziale P.)
 - Kognitive Beschränkungen
 - Mängel der Urteilskraft
 - Kommunikation, Koordination
 - Gruppendynamik
 - Verborgene Ziele
 - Fehler

Welt der Lösungsansätze:

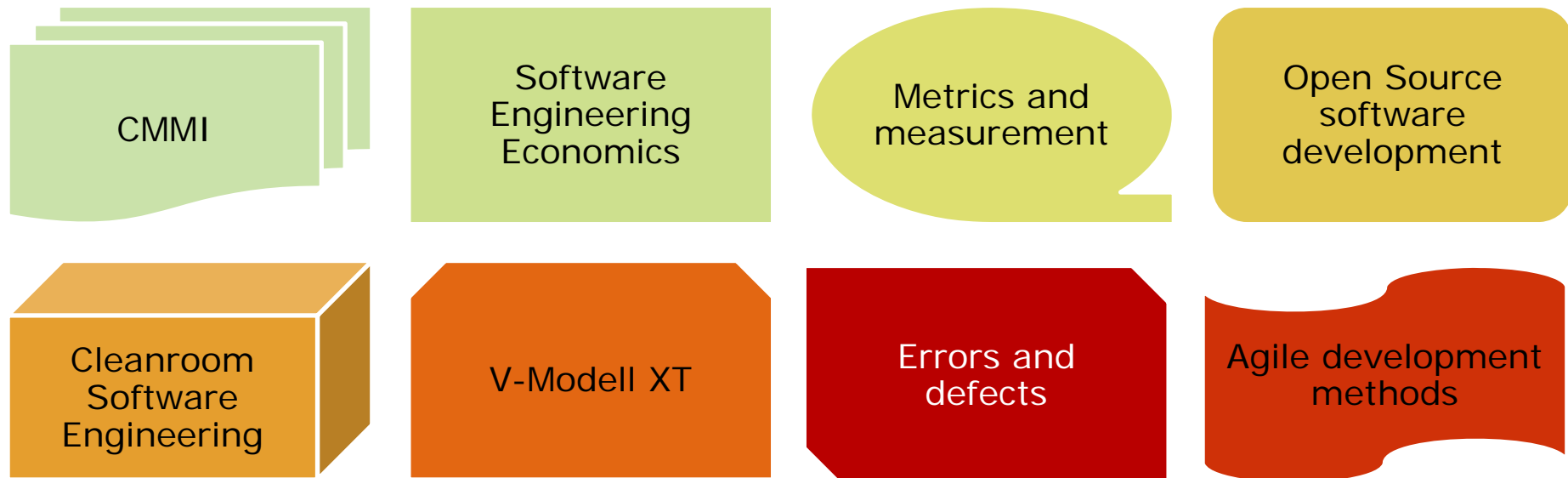
- Technische Ansätze ("hart")
 - Abstraktion
 - Wiederverwendung
 - Automatisierung
- Methodische Ansätze ("weich")
 - Anforderungsermittlung
 - Entwurf
 - Qualitätssicherung
 - Projektmanagement

- Walk through the topics of this course
 - What is it about?
 - Why is it of interest?
 - What will we look at?
- Walk through the topic areas of Vorlesung "Softwaretechnik"
 - What has been discussed in "Softwaretechnik" (SWT)?
 - What has not?
 - How does that relate to the contents of this course?

Agenda (2)

- Walk through the topics of this course
 - What is it about?
 - Why is it of interest?
 - What will we look at?

Topics:



What is it about?

- A description of the process capabilities required for producing systems efficiently and reliably
 - and of the order in which best to achieve them



Why is it of interest?

- CMMI is the most well-known framework for process improvement

What will we look at?

- Base ideas
- Process domains covered
- Levels and key process areas; some example practices

	Level	Capability	Result
5	Continuous Process Improvement Optimizing	Organizational Innovation & Deployment Causal Analysis & Resolution	Productivity & Quality
4	Quantitatively Managed	Quantitative Process Management Software Quality Management	
3	Defined Process Standardization	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Product Management Risk Management Integrated Teaming Integrated Supplier Management Decision Analysis & Resolution Organizational Environment for Integration	
2	Managed Basic Project Management	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Measurement & Analysis Product & Process Quality Assurance Configuration Management	
1	Initial Heroic Efforts	Design Develop Integrate Test	
			Risk & Waste

What is it about?

- Judge a software process not just by cost and quality, but also take the value generated into account

Software
Engineering
Economics

Why is it of interest?

- That is what engineering should be about!

What will we look at?

- Conventional view vs. economical view
- Tracking earned value
- Design decisions as buying real options

Metrics and measurement

What is it about?

- Measurement is assigning a symbolic value to an object in order to characterize a certain attribute of that object

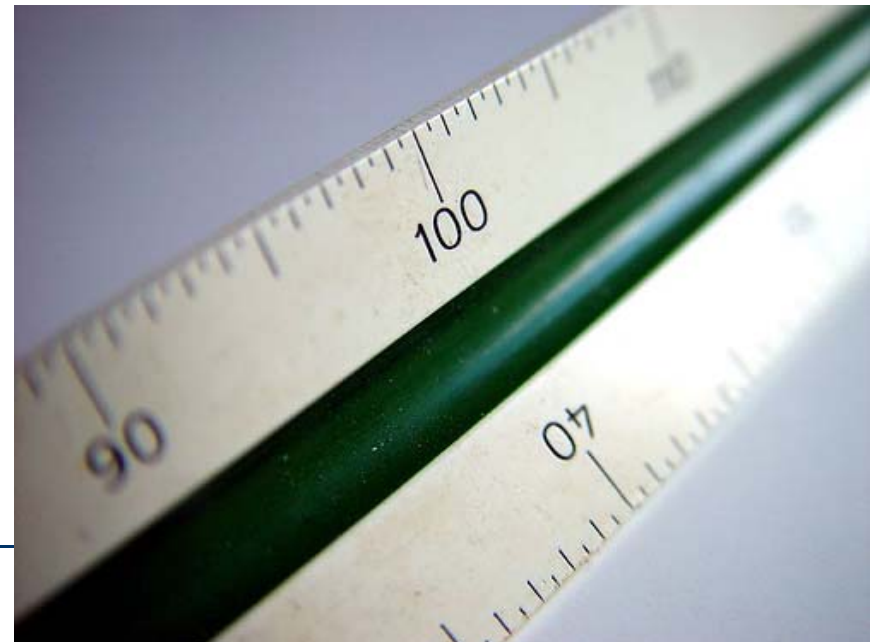
Metrics and measurement

Why is it of interest?

- Sometimes allows talking about a complex situation in a way that is simple, yet precise and objective

What will we look at?

- Base ideas (measurement, scale type, validity, inference)
- Common mistakes
- Example metrics (product, process)



Open Source software development

What is it about?

- Developing free-to-use software with teams of volunteers

Open Source
software
development

Why is it of interest?

- High impact and popularity of some of the resulting software
 - High relevance as competition of commercial SW development
 - High relevance as a pillar of various business models
- Similar to commercial distributed development

What will we look at?

- Foundation factors (motivational, business)
- Approaches used for requirements, design, quality assurance, project management



Cleanroom Software Engineering

What is it about?

- A development process for systems requiring very high reliability



Why is it of interest?

- It focuses on defect prevention rather than detection/removal
 - like the clean rooms in chip manufacturing

What will we look at?

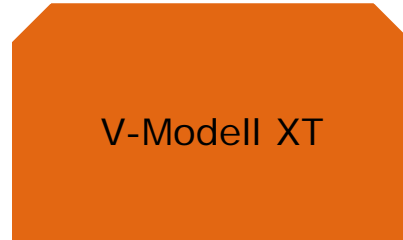
- quasi-formal specification
- successive refinement
- correctness arguments
- statistical testing



V-Modell XT

What is it about?

- A comprehensive, tailorable process model



Why is it of interest?

- Standard process model for German government projects
 - also popular elsewhere

What will we look at?

- Basic approach
- Main elements
- Tailoring



Errors and defects

What is it about?

- Any defect in software is a consequence of an error or a mistake. Why do they occur and how can they be prevented?

Errors and defects

Why is it of interest?

- Most time in most software processes is spent committing errors or detecting, locating, and repairing defects

What will we look at?

- Definitions of "error", "defect" etc.
- Classifications of errors and of defects
- Reasons for errors
- Error prevention



Agile development methods

What is it about?

- Process models that focus on quick generation of end-user value rather than on planning and documentation



Why is it of interest?

- Well-suited for many projects
 - typically smaller or information-system projects

What will we look at?

- Base ideas, common misunderstandings
- examples: eXtreme Programming, Scrum, Kanban, ...

Agenda (3)

We will now walk through the topic areas of SWT
(Vorlesung "Softwaretechnik"):

- Describe what has been discussed in SWT
- Describe what has not
- Describe what will be discussed in this course

Requirements
Engineering
("Anforderungs-
bestimmung")

Design
("Entwurf")

Constructive
quality assurance
/ Process
management

Analytical quality
assurance

Project
management

Reuse ("Wieder-
verwendung")

Covered in SWT:

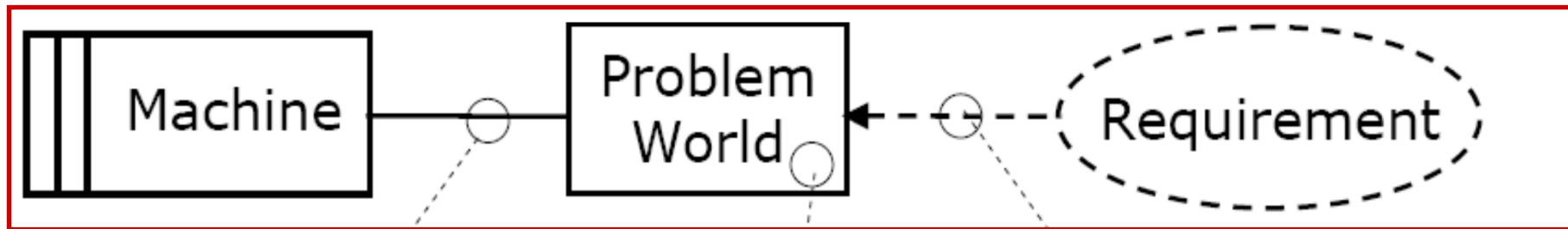
- Fundamentals
- Elicitation techniques
- Specification technique: Use Cases
- Specification technique: UML + OCL

Requirements
Engineering
("Anforderungs-
bestimmung")

Not covered in SWT (nor here):

- Elicitation techniques in detail
 - detailed methodology, problems, case studies
- Other formal specification techniques
 - for embedded or high-assurance systems: Z, VDM, CAML, etc.
- Analysis techniques
 - detecting incompleteness, detecting inconsistency
- Requirements management techniques
 - arbitration, change management etc.

RE: The grand view



We will build the machine to behave like this, so that ...
(Software Specification **S**)

© Michael Jackson

... knowing that the problem world is like this, ...
(Problem World Properties **W**)

... we'll be sure the resulting effects in the problem world will be these
(Requirement **R**)

- Software Engineering must cover all three
 - because formalizing W and R is difficult and
 - because it is unclear what part of the infinite W is relevant

RE: in this course

- We will not have Requirements Engineering (RE) as a chapter headline in this course
- But we will talk about RE in the context of agile development methods and Open Source SW dev.
 - and will learn about their integrated approach towards specifying S, W, and R

Covered in SWT:

- Fundamentals
- Modularization
- Architectural styles
- A few basic design patterns

Design
("Entwurf")

Not covered in SWT:

- Reference architectures in depth
- Many more design patterns
 - e.g. for distributed systems, middleware, component technologies, real time systems, high-assurance systems
- Evaluating modularization
 - → chapter on Software Engineering Economics

Design: in this course

- We will not have Design as a chapter headline in this course
- But we will talk about design in the following contexts:
 - Quantifying the value of modularity
 - The design process of Open Source Software development
 - The design approach of Cleanroom Software Engineering
 - The design approach of agile methods
 - The nature of design errors

Constructive quality assurance / Process management

Covered in SWT:

- Fundamentals
- Basics of process management frameworks
- Basics of some process models

Constructive
quality assurance
/ Process
management

Not covered in SWT:

- Details of process management frameworks
 - → chapter on CMMI (Capability Maturity Model Integration)
- Details of process models
 - → chapter on V-Modell XT
 - → chapter on Cleanroom Software Engineering
 - → long chapter on agile process models
 - → long chapter on Open Source development
- Defect prevention
 - → chapter on errors and defects

Process mgmt: in this course

- By and large, most parts of this course use process management as their main perspective
- However, it is only used as a perspective, not as a restriction: the actual contents will be broad.

Analytical quality assurance (QA)

Covered in SWT:

- Fundamentals
- Basic techniques for
 - choosing test inputs,
 - choosing test objects,
 - determining correct outputs,
 - regression testing,
 - test termination
- Static analysis: Review, automated analysis

Not covered in SWT (nor here):

- Detailed techniques for all of the above



Analytical quality
assurance

Analytical QA: in this course

- We will not have Analytical Quality Assurance as a chapter headline in this course
- But we will talk about it in the following contexts:
 - The QA approach of agile methods
 - The QA approach of V-Modell XT
 - The QA approach of Cleanroom Software Engineering
 - QA issues in Open Source Software development
 - Reliability modeling (as part of the Cleanroom method)
 - Errors and defects

Project management

Covered in SWT:

- Fundamentals
- Basics of estimation, planning, coordination, communication, non-linear dynamics
- Social psychology, personality types



Project
management

Not covered in SWT (nor here):

- Details of estimation techniques
- Systems theory and non-linear dynamics
- Advanced planning, coordination, communication
- Advanced people issues

Project management: in this course

- We will not have Project Management (PM) as a chapter headline in this course
- But we will talk about it in the following contexts:
 - The role of PM in CMMI
 - The PM approach of agile methods
 - The PM approach of V-Modell XT
 - PM issues in Open Source Software development

Covered in SWT:

- Fundamentals
- Analysis patterns, design patterns, process patterns, usability patterns, anti-patterns

Reuse ("Wieder-
verwendung")

Not covered in SWT (nor here):

- More patterns
- Library of other reusable things (e.g. document templates)

Reuse: in this course

- Reuse is a background idea of all of software engineering
- It was a recurring topic throughout much of SWT
 - "Normal design"
- It is implicitly a constant topic throughout this course
 - Reusable software *process* concepts

Thank you!