



Stabilitätspakt für Südosteuropa
Gefördert durch Deutschland
Stability Pact for South Eastern Europe
Sponsored by Germany



Quality Assurance of the JCSE Materials: Extensions and Modifications



Klaus Bothe

7th Workshop "Software Engineering Education and Reverse
Engineering" Risan, Montenegro, September 2007



**Modification and extension of
JCSE teaching materials done as
part of the preparation and the
delivery of the JCSE in Tirana in
March 2007**



Some problems with JCSE materials

- Generation of pdf-files from ppt-files for students: time-consuming, error-prone (wrong)
- Case study "SemOrg" (Requirements specification): tables no more readable, wrong layout in case of enumerations, figures not readable ...
- Many topics were translated by means of electronic German-English-Dictionaries:
Slides never subject to quality assurance (correct language, correct translation?)
- Not all topics available
- Useful materials missing: examination questions, handouts ...

However, all in all: slides in a rather good state ...

3



Agenda

- New documents
- Changes
- Statistics

4

What has been done:


new documents (teaching materials) added

- **Printouts:** pdf version of all slides for students:
 - 1sided (colored for viewing),
 - 4sided (black-white for printing)
- **Handouts:** File with a collection of 48 fundamental slides
- **Examination questions**
- **Alternative implementation:**
Case study SemOrg


5

Joint Course in Software Engineering
- subproject of: "Software Engineering: Computer Science Education and Research Cooperation" -

Introduction



Stabilitätspakt für Südosteuropa
Gefördert durch Deutschland
Stability Pact for South Eastern Europe
Sponsored by Germany



Deutscher Akademischer Austausch Dienst
German Academic Exchange Service

Joint Course in Software Engineering (JCSE) is the part of the project "Software Engineering: Computer Science Education and Research Cooperation" that is supported by DAAD under auspices of Stability Pact for South-Eastern Europe sponsored by Germany. The aim of this subproject is to define a common curriculum for the undergraduate course software engineering that would be used (partly or as a whole).

New documents
(teaching materials)
added

This site contains only material that is directly related to this subproject. The material is generally divided into two groups (see menu below):

- project management - related to administration and management of the project.
- course materials - results of the project.

For further information on the whole project, click [here](#).

© 2001 - 2006. For details of the project please contact [Klaus Bothe](#). For the site contents contact [Zoran Budimas](#).

Project management				Course Materials			
Project schedule	Basic principles	Review reports	F.A.Q.	Structure of the course	Topics	Slides	Assignments
Responsibilities (roles)	Slide style guides	Slide version management	Copyright policy	Case studies	Literature	Slide printouts	Examinations
Workshop presentations				Tools	Demo topics (ppt.doc)	Slide handouts	Students feedback form

New documents added:
Printouts (pdf)

Joint Course in Software Engineering

- subproject of: "Software Engineering: Computer Science Education and Research Cooperation" -

Slide Printouts in pdf format

1s means '1 slide per page, coloured'
4s means '4 slides per page, black/white'

PART I

- 1 - What is Software engineering?
[Topic01-2007-02-16-1s.pdf](#), 1,00 MB; [Topic01-2007-02-16-4s.pdf](#), 1,24 MB
- 2 - Quality criteria for software products
[Topic02-2007-03-13-1s.pdf](#), 0,18 MB; [Topic02-2007-03-13-4s.pdf](#), 2,00 MB
- 3 - Software process models - introduction
[Topic03-2007-02-17-1s.pdf](#), 0,66 MB; [Topic03-2007-02-17-4s.pdf](#), 0,61 MB
- 4 - Basic concepts and software development documents
[Topic04-2007-02-26-1s.pdf](#), 1,06 MB; [Topic04-2007-02-26-4s.pdf](#), 0,32 MB

Used as material
for students
(ppt for lecturers)

7

New documents added: Handouts
(important slides to be given to the students before the lectures)

JCSE "Software Engineering"

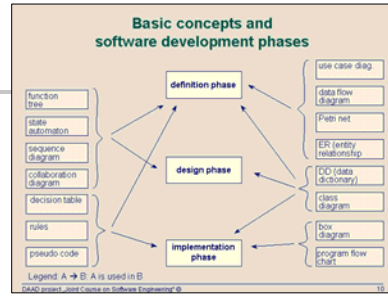
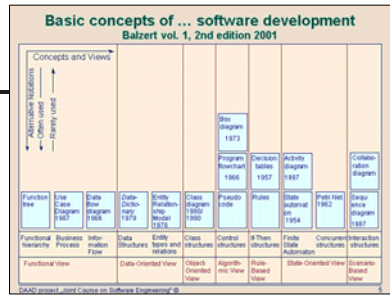
Version: March 2007

Handouts

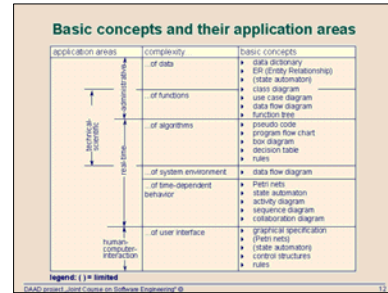
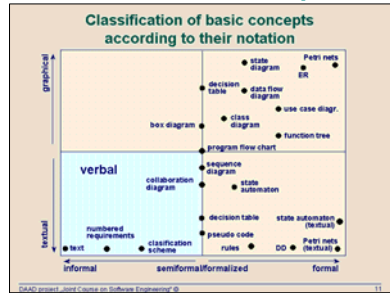


Important slides (48)
to be given to students
before the lectures

8



Handouts: topic 4



9



New documents added: Examination questions

Joint Course in Software Engineering

- subproject of: "Software Engineering: Computer Science Education and Research Cooperation" -

Examinations

These are examples of examination questions

The first one is a collection used in Novi Sad.
The other ones had been applied in a course at Polytechnic University Tirana in June 2007.

- All of the questions (.doc, 0,83 MB)
- Part1:General questions (.doc, 0,63 MB)
- Part2:Testing and Software Metrics (.doc, 0,70 MB)

10

Examination questions: example of metrics

Software Metrics:

1. Basics of software metrics:
 - a) What is a software metric?
 - b) Why should software be measured?
 - c) List some software metrics – classified by phases of software development.
2. Cyclomatic complexity (general questions):
 - a) What is the basic goal of measuring cyclomatic complexity?
 - b) What is the basic definition of cyclomatic complexity.
 - c) Which maximum limit is recommended for cyclomatic complexity?
3. Cyclomatic complexity (example)

Given the following Java program code (sequence of statements):

```
if (x > y)
    max = x;
else
    max = y;
if (x < level)
    print("error"); //print defined somewhere else
level = x;
```

 - a) Draw the control flow graph CFG of these statements on a separate paper (Attach corresponding program code to the nodes of the graph)
 - b) Based on that control flow graph: What is the cyclomatic complexity of the Java program?
4. Enumerate at least 5 factors that style metrics measure.

11

Agenda

- New documents
- Changes
- Statistics

12

What has been done: *existing* slides changed and extended

Reasons for slide changes:

- New master slide: logos
- New slides added
- Contents moved
(exchange of slide order, move from other topics: V model)
- Slide contents extended
- Improved layout of slides
- Misspellings corrected
- Wrong translations, wrong contents corrected
- German terms not translated
(e.g. German 'Automat' → automaton)
- Mixture of English and American:
cancelling (British) – canceling (American)
- Animation errors
- Style guide errors
- Lecture notes extended

13

New JCSE topic title slide

Topic 3
Software process models

DAAD Project
"Joint Course on Software Engineering"

Humboldt University Berlin, University of Novi Sad, University of Plovdiv,
University of Skopje, University of Belgrade, University of Niš, University of Kragujevac,
University of Zagreb, University of Timisoara, University of Tirana, University of Sarajevo,
University of Banja Luka, University of Rijeka, Polytechnic University Tirana

Version: Mar. 09, 2007 (D Apr. 23, 2006)

14



Slides: contents moved

**Topic 19:
Testing**

**Topic 3: Software
process models**

V - Model

**RUP
(incremental process)**

Reason: all models concentrated in topic 3

15



Slides: new ones added

**Example: 5 new slides in topic 1
(Introduction to Software Engineering)**

→ Next slides

16

Süddeutsche Zeitung
23./24. Okt. 2004

▶ Software-Entwickler (m/w)

Dipl.-Informatiker/Dipl.-Ing.
Kennziffer SZ/401, mehrere Positionen zu besetzen

Ihre Aufgabe

- ▶ Architekturentwurf, Systemdesign und Softwareentwicklung für Mikroprozessor-Systeme im Bereich der Prozeßautomatisierung
- ▶ Inbetriebnahme und Funktionstest bis zur Serienreife

Ihr Profil

- ▶ abgeschlossenes Studium der Fachrichtung Informatik, Softwaretechnik, Elektrotechnik o.ä.
- ▶ mehrjährige Berufserfahrung in der Softwareentwicklung für Steuerungstechnik
- ▶ Erfahrung in der Programmierung mit C, C++ und Assembler sowie im Software-Engineering
- ▶ Freude am selbständigen, eigenverantwortlichen Arbeiten in Projekten

Software developer

Your task:

- Architectural design, system design ...
- Installation and functional test ...

Your profile:

- Completed study
- experience in software development
- experience with programming in C++ and in software engineering
- independent work
- good English

SWEBOK: Software Engineering Body of Knowledge

Guide to the SWEBOK®

Home | Contact Us

Franglais

Español

Download the latest Version (Feb. 16, 2005)

Also available in book format

First International Workshop (Jul. 25-28, 2005)

Reviewer Demographics

Reviewer Response Database Search Tool

Project Overview

Project Contributors

A Three-Phase Approach

- Straw Man
- Stone Man
- Iron Man

Available Documents

The First International Workshop on the Evolution of the Guide to the Software Engineering Body of Knowledge In Conjunction with COMPSAC 2005

Edinburgh, Scotland, July 25-28, 2005

Guide to the Software Engineering Body of Knowledge

A project of the

Software Engineering Coordinating Committee

Project managed by

Ecole de technologie supérieure

The following motion was unanimously adopted by the Industrial Advisory Board on February 6, 2004.

The Industrial Advisory Board finds that the Software Engineering Body of Knowledge project initiated in 1998 has been successfully completed, and endorses the 2004 Version of the Guide to the SWEBOK and commends it to the IEEE Computer Society Board of Governors for their approval.

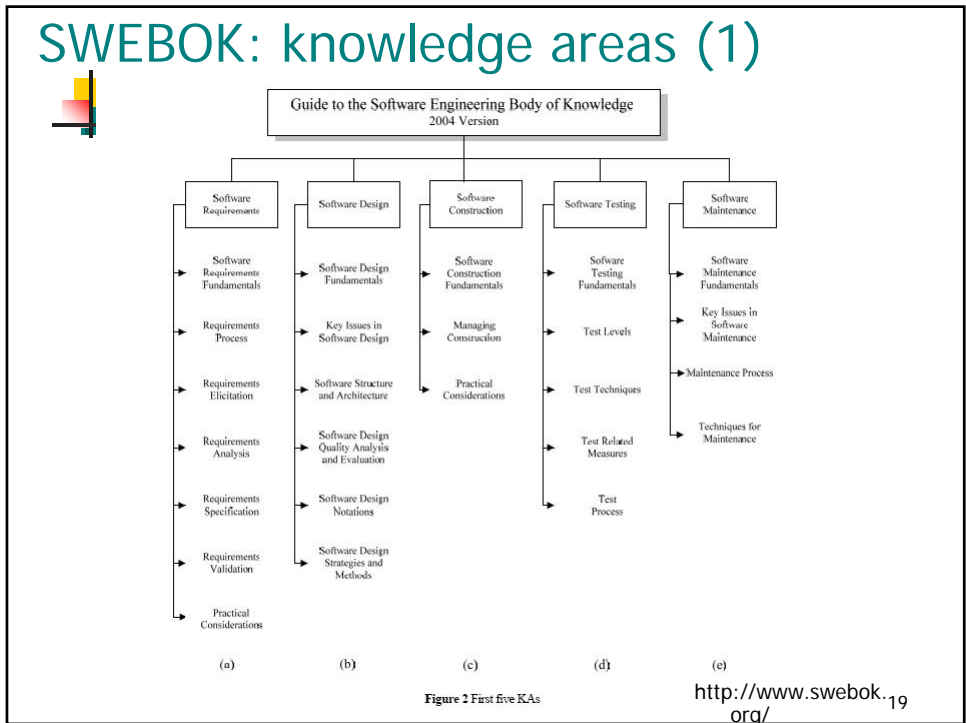
Corporate support by

CANADIAN COUNCIL OF PROFESSIONAL ENGINEERS
CONSEIL CANADIEN DES INGÉNIEURS

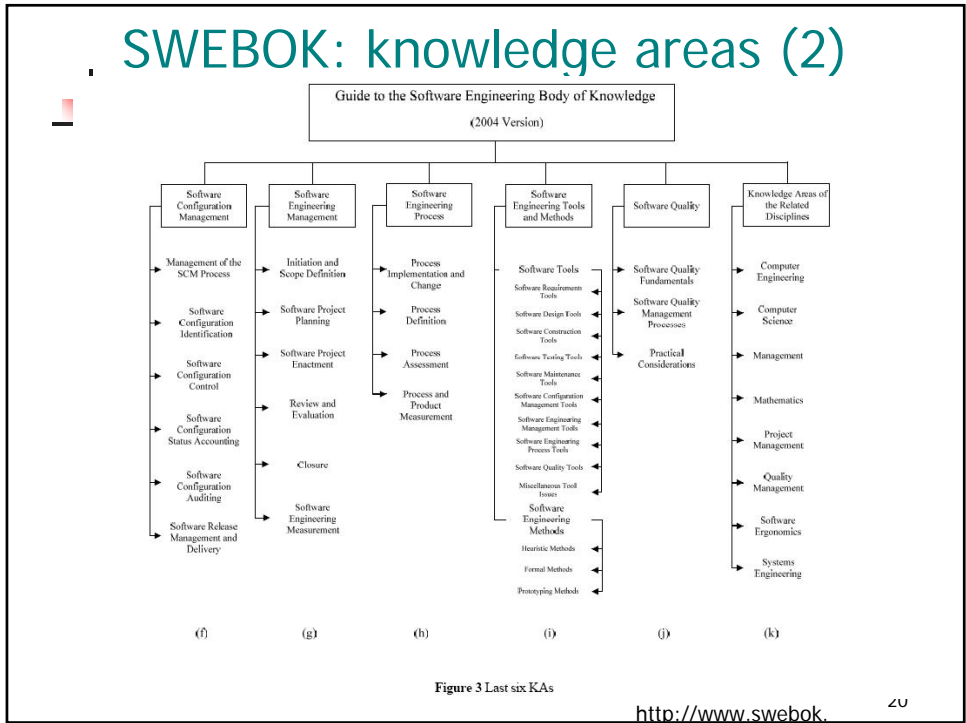
National Research Council Canada
Conseil national de recherches Canada

http://www.swebok. 18

SWEBOK: knowledge areas (1)



SWEBOK: knowledge areas (2)





SE: teaching materials in the internet

- Swenet (<http://www.swenet.org/>)
- Musoft (<http://www.musoft.org/>)
- Ariadne (<http://www.ariadne-eu.org>)
- Merlot (<http://www.merlot.org>)
- JCSE (our DAAD project)
- MIT (top university in US)

21



Slides: contents on a slide extended

Example: Next slide (topic 1)
→ Box added

22

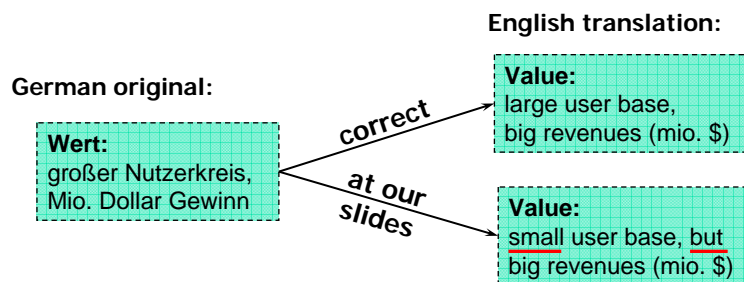
How large is software ?

- **Classification of software according to its size:**
small: up to 2000 lines of code
middle: 2000 – 100.000
large: 100 000 – 1.000.000
very large: > 1 Mio.
(A. Macro, p. 70)
- **Average size of software in each of the 100 largest US companies:**
35 Mio. lines of code
(Hausi Müller, p. 3-12)
- **e.g. USA, DoD:**
1,4 billion lines of code
(distributed to 1.700 data centers)
operating costs for these systems:
9 billion dollar / year
(CACM, 5, 1994, p. 26)

CSEET 2001, Charlotte, USA:
"In Europe, nobody can imagine how developed software engineering in the US is - thanks to military industry."

23

Translation from German to English not exact



„Restoring a Legacy: Lessons learned“
(Nortel experience report,
Software Engineering, Selected Readings,
IEEE 2000 und IEEE Software July 98)

24



Slides: misspellings

A lot of misspellings:
see statistics

25



Agenda

- Problems
- Changes
- Statistics

26

Some statistics about modifications in 2007

For 491 (out of 1200 existing) slides:
at least 73 documented modifications (15 %)
/ estimated 120 (25 %)

- New slides added: 17
- Slide contents extended: 25
- Improved layout of slides: 11+
- Misspelling: 6+
- Wrong translations, wrong contents: 15+
- German terms not translated (e.g. German 'Automat' → automaton): 4+
- Animation errors: 2++

27

Incomplete statistics: at least 15 % of all slides modified in 2007

Topic	New Slides	Extended slides	Layout improved	Misspellings	Wrong translations	German not translated	Animation error	CHANGES (no. slides)
1	6	3	?	?	3			12 (62)
2		?						? (19)
3	8	4	3		-	-	?	17 (53)
4	1	1	1					3 (19)
5	-	1	3			?		4 (51)
6								? (64)
7	2	3	2	1				8 (45)
10		1			4			7 (51)
11	-	8	2	3	4	4	2	23 (54)
12	?							? (18)
13	-	4	-	2	4			10 (55)
Sum	17 (491)	25 (491)	11+ (491)	6++ (491)	15+ (491)	4+ (491)	2++ (491)	73+ (491)

Consequence of modifications?

- Use only of English slides: no problem
- Slides translated to national language: see lecture notes of title slide for documented modifications (time-consuming, documentation complete?)
→ tool-supported determination of slide differences necessary

29

Four JCSE topics not available (as planned in the beginning, as part of JCSE)

in minutes	Z.B. (W 02)	K.B. (S 03)		
Part I: Introduction			Part III: Software Design	
• 1. What is software engineering	80	120	• 15. Overview of design activities	-- 90
• 2. Quality criteria ...	40	45	• 16. Structured design	-- 15
• 3. Software process models	120	90	• 17. Object-oriented design	-- 45
• 4. Basic concepts ...	60	40	Part IV: Implementation and testing	
Part II: Requirements engineering			• 18. Implementation	-- 90
• 5. Results of the ... phase			• 19. Systematic testing	-- 180
• 6. Cost estimation			Part V: Advanced problems	
• 7. Function-oriented view	60	50	• 21. Software metrics	-- 180
• 8. Data-oriented view	50	35	• 22 Maintenance	-- -
• 9. Rule-oriented view			• 23. Reverse engineering	-- 90
• 10. Structured analysis	80	65	• 24. Quality of software development ...	-- 90
• 11. State-oriented view	(45)		• 25 Software ergonomics	-- 180
• 12. Scenario-oriented view			• 26 User manuals	- -
• 13. Object-oriented analysis			• 27 Project management	? 90
• 14. Formal software specification ...		150	• 28. Configuration ... management	- 45

Instead of this: additional topics had been produced