Karlsruhe Institute of Technology

Herzlich willkommen!

Softwaretechnik II

JProf. Dr. Oliver Hummel, Vertretungsprofessor, IPD

Topic 1

Introduction

SOFTWARE DESIGN AND QUALITY GROUP
INSTITUTE FOR PROGRAM STRUCTURES AND DATA ORGANIZATION, FACULTY OF INFORMATICS

sdq.ipd.kit.edu

Karlsruhe Institute of Technology

Herzlich

n!

Softw

JProf. D

Topic 1

Introduc

SOFTWARE
INSTITUTE F

kit.edu

Overview for Today

- Content
  - a few words about me
  - a few more words more about the course
  - challenges in software development
  - and some ideas to overcome them
    - i.e. the content overview for this semester

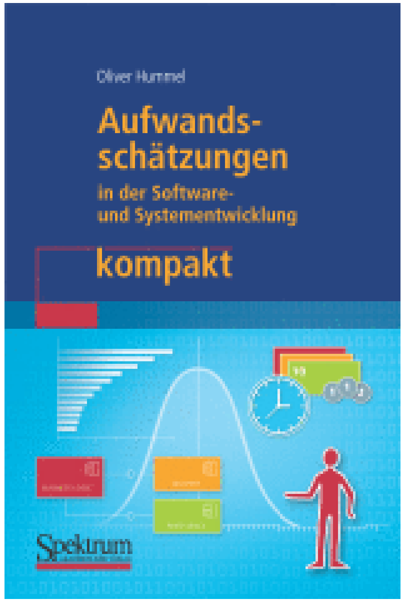
- Learning goals for today, students –
  - have a coarse overview of the course content
    - and understand why it could be important for them
  - have refreshed some core challenges of software development
    - and have an appetite for learning how to overcome them

About Me

- *Good mix of* ***academic*** *…*
  - Juniorprofessor at U Mannheim since Feb 2010
    - currently on leave
  - PhD in Software Engineering from U Mannheim, 2008
  - Teaching assignments from IU Bruchsal and HS Mannheim
  - Diploma from TU Kaiserslautern in 2003

- *… and* ***practical*** *experience*
  - projects with EADS and the German Navy
  - SE Consulting for Dell (f.k.a. Perot Systems)
    - projects for Lufthansa, Henkel and others

- *Selected highlights*
  - book on cost estimation published April 2011
  - University Teaching Certificate (July 2012)



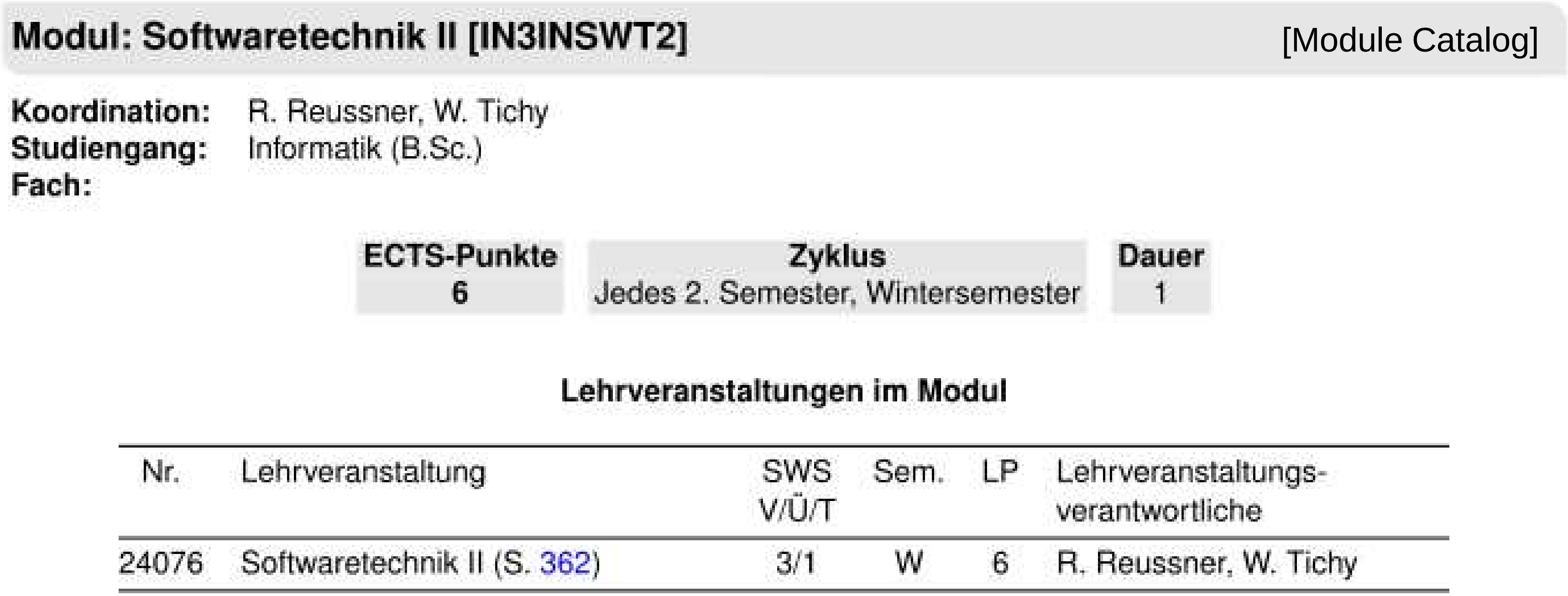
SWT II – at a Glance

- Goal
  - deepen your understanding of systematic software development
    - by discussing selected state of the art approaches
  - develop a critical understand of principles, methods and tools

- Lecture Materials
  - available through the study portal
    https://studium.kit.edu/sites/VAB/0x667A65286804974D85DD99B4879E9E22
  - **Password: ipdswtii**

➔ merely slides are provided
  - are neither intended to replace attendance or notes
  - nor reading additional literature
    - pointers usually at the end of the slides

More about SWT II (1)



Modul: Softwaretechnik II [IN3INSWT2]

[Module Catalog]

Koordination: R. Reussner, W. Tichy
Studiengang: Informatik (B.Sc.)
Fach:

ECTS-Punkte	Zyklus	Dauer
6	Jedes 2. Semester, Wintersemester	1

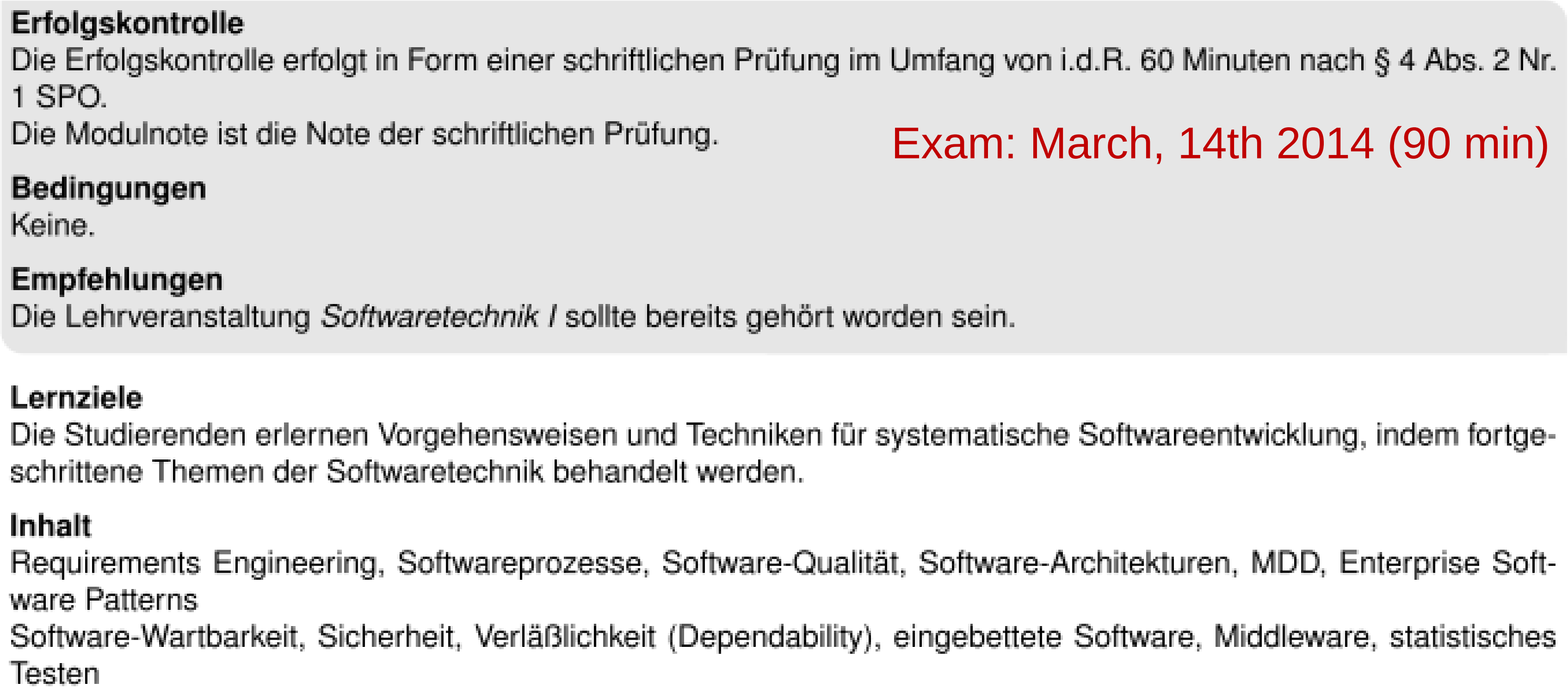
Lehrveranstaltungen im Modul

Nr.	Lehrveranstaltung	SWS V/Ü/T	Sem.	LP	Lehrveranstaltungs-verantwortliche
24076	Softwaretechnik II (S. 362)	3/1	W	6	R. Reussner, W. Tichy

180 Zeitstunden

➔ *some slides are courtesy of Prof. Reussner*

More about SWT II (2)



Erfolgskontrolle

Die Erfolgskontrolle erfolgt in Form einer schriftlichen Prüfung im Umfang von i.d.R. 60 Minuten nach § 4 Abs. 2 Nr. 1 SPO.

Die Modulnote ist die Note der schriftlichen Prüfung.

Bedingungen

Keine.

Empfehlungen

Die Lehrveranstaltung *Softwaretechnik I* sollte bereits gehört worden sein.

Lernziele

Die Studierenden erlernen Vorgehensweisen und Techniken für systematische Softwareentwicklung, indem fortgeschrittene Themen der Softwaretechnik behandelt werden.

Inhalt

Requirements Engineering, Softwareprozesse, Software-Qualität, Software-Architekturen, MDD, Enterprise Software Patterns

Software-Wartbarkeit, Sicherheit, Verläßlichkeit (Dependability), eingebettete Software, Middleware, statistisches Testen

Exam: March, 14th 2014 (90 min)

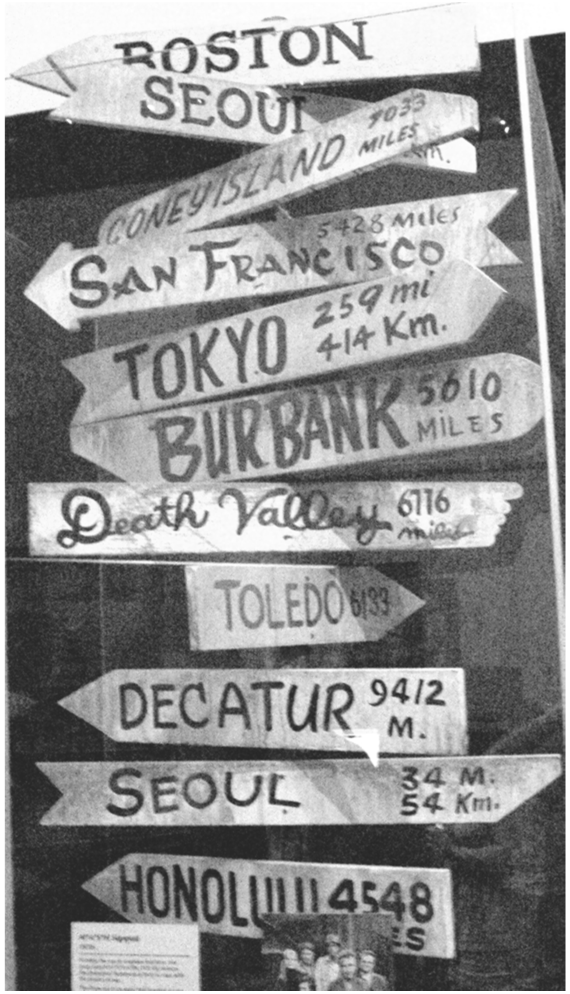
SWT II is …

… important for your future career

- as will be demonstrated in a second

… a signpost for later specialized lectures in the area of software engineering

- Empirische Software-Technik
- Multikern-Rechner und Rechnerbündel
- Software-Evolution
- Komponentenbasierte Software-Architektur
- .NET
- Performance of Enterprise Applications
- Modellgetriebene Software-Entwicklung
- ...



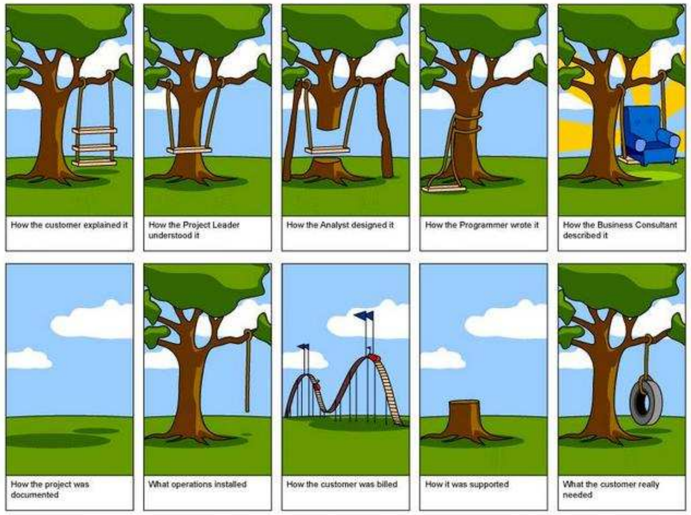
[Wikipedia]

Skill Targets (Kompetenzziele)

- The completion of this course enables students –
  - to **understand** the benefits of state of the art software engineering approaches/techniques
  - to **select** an appropriate approach/technique for a given purpose/problem
  - and ideally **apply** it in practice
    - perhaps with some additional training

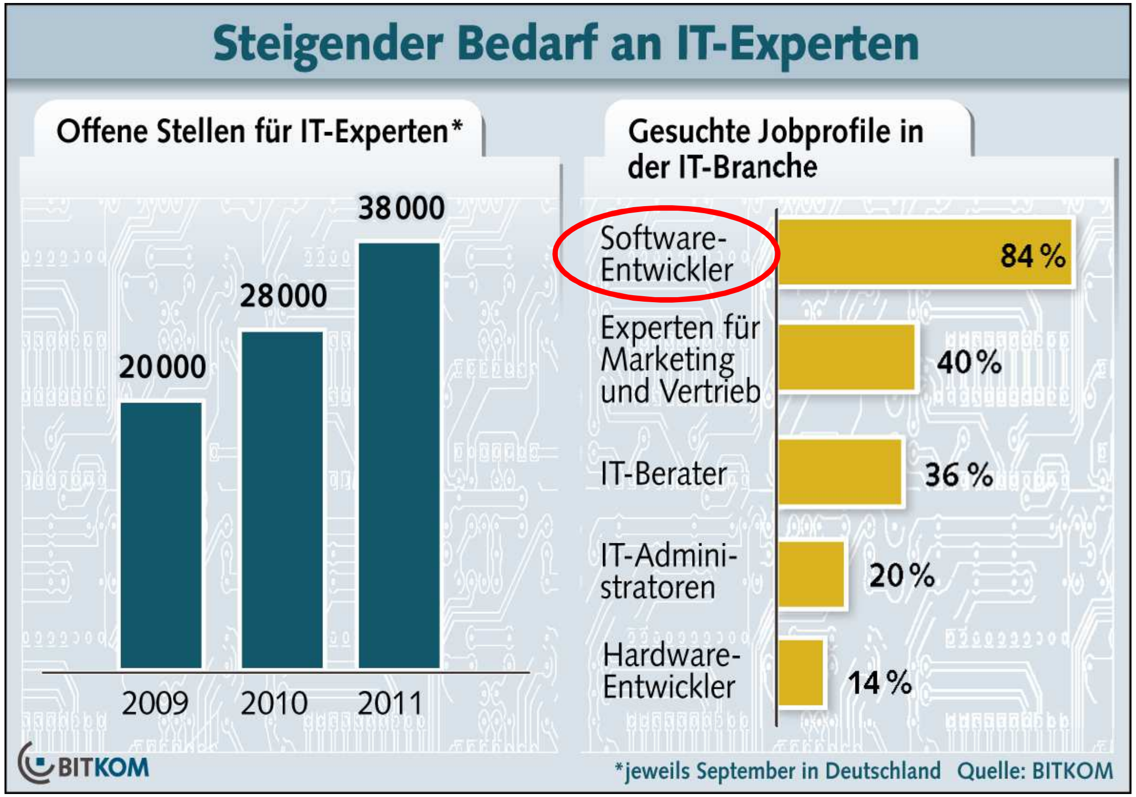
  ➔ *so they can become productive in a software development project quickly*

- In a nutshell, students are provided with the basic terminology of **“geek speak”**
  - *and are able to avoid situations like these →*


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

Career Outlook

- BITKOM surveys regularly underline a lack of IT specialists
  - at least ***38,000 open positions*** in Germany alone



[http://www.bitkom.org/de/presse/8477_69920.aspx]

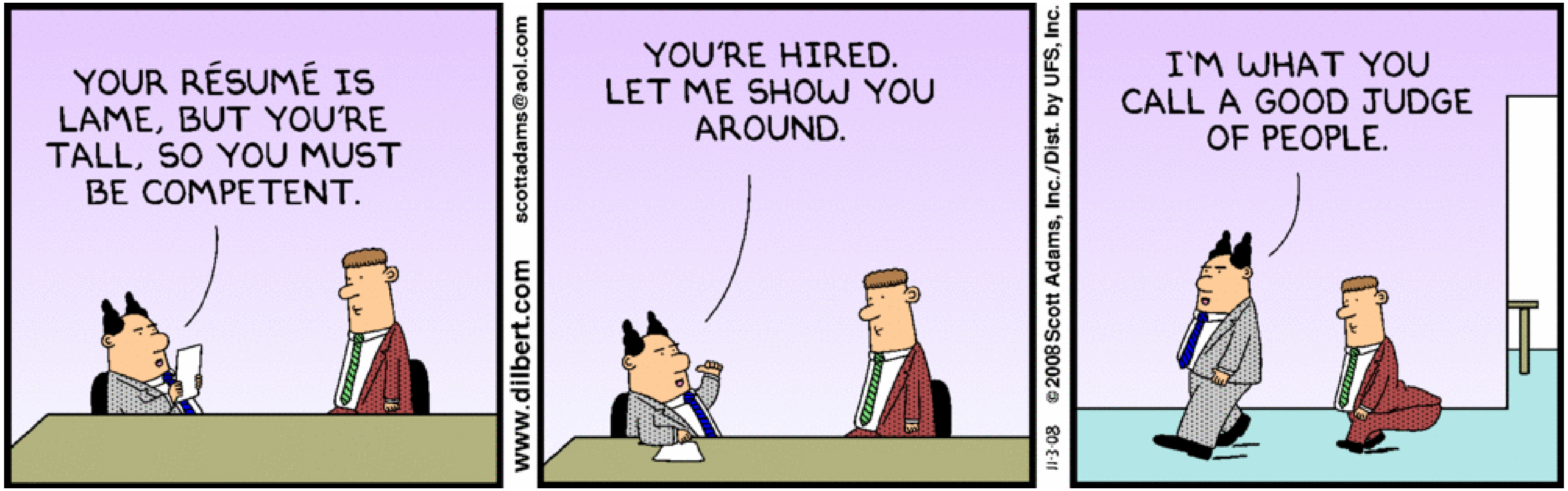
For (soon to be) Master Students

- You may want to document your background regarding the Enterprise Software of tomorrow via an additional certificate

  „Software Innovations for the Digital Enterprise“

  - issued by the Software-Cluster
    - *einem Spitzencluster des BMBF*
    - when at least 20 (3..6) credits from the following areas are collected –
      - Software Engineering
      - IT Security
      - Cloud Computing
      - Pervasive Computing
  - cf. http://www.software-cluster.org/de/karriere/qualifizierung/zusatzqualifikat-im-master

Grow with your Challenges …


YOUR RÉSUMÉ IS LAME, BUT YOU'RE TALL, SO YOU MUST BE COMPETENT.
scottadams@aol.com
www.dilbert.com
YOU'RE HIRED. LET ME SHOW YOU AROUND.
© 2008 Scott Adams, Inc./Dist. by UFS, Inc.
11-3-08
I'M WHAT YOU CALL A GOOD JUDGE OF PEOPLE.

[dilbert.com]

Learn Different.

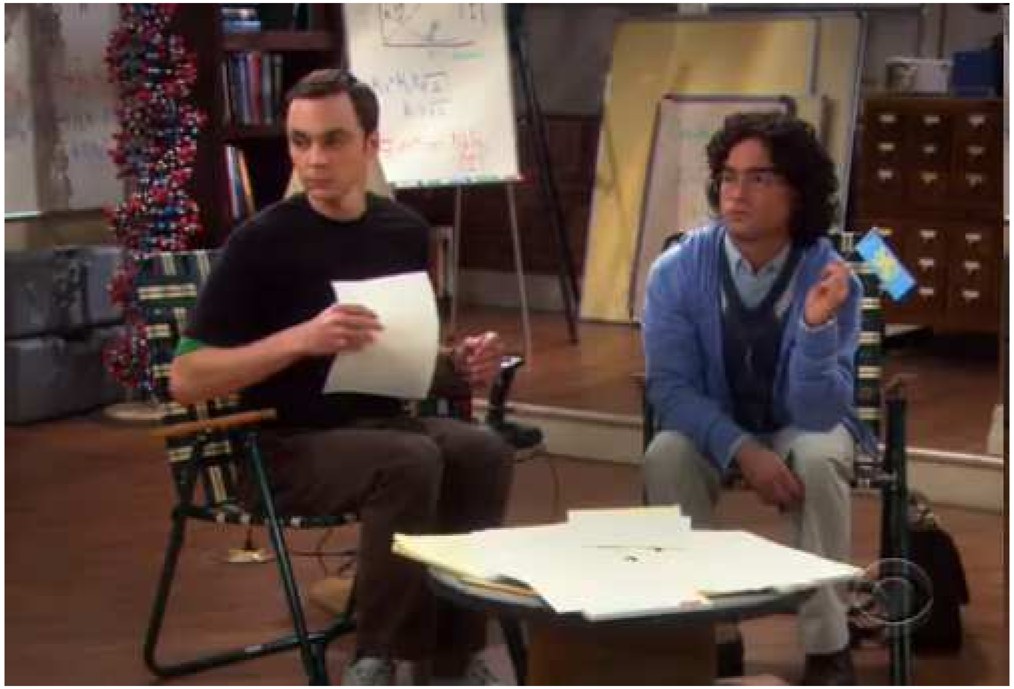
- A word on ***activating teaching methods*** and the learning environment
  - knowledge cannot be „beamed“ from my head into yours
    - as shown by research in didactics

➔ Rather, it is important that you engage yourself with the content of the course

- I will offer the environment to do this in the classes
  - this might initially feel a bit different for most of you
  - however, you are not obliged to participate
  - feel free to use other ways of learning if they also suite you
    - or better: use my offerings and your approach together

➔ **We will try that later …**

On Roommate Agreements



- http://www.youtube.com/watch?v=N0qDy0T5WXM

Our „Roommate“ Agreement

- Some basic proposals for a successful and enjoyable collaboration during the semester

Lecturer	Students
The lecture begins and ends on time	Students appear on time and stay until the end
The lecture presents the state of the art in research and practice	Students fill potential gaps regarding fundamentals independently
The lecturer is always well prepared	The students prepare and rework the lectures/tutorials as requested
The lecturer uses activating teaching methods	Students cooperate actively and constructively *(or stay at least quite during them)*
The lecturer is clearly audible and schedules time for questions	Students have no side conversations and use WWW only on request

Further Services

- Further offers for the students, the lecturer –
  1. is open for special questions after the lecture
     - ***and offers a consultation hour***
       - one hour on Tuesdays from 2:00 pm
       - or upon request
  2. answers questions via e-mail
     - if the answer is relevant for everybody it will be made available in the course materials
  3. makes materials for the course available online
     - the slides are usally made available on time after the lecture
  4. makes the evaluation criteria transparent
     - i.e. provides students with exercises illustrating potential exam questions
       - *last exams are available from the Fachschaft*

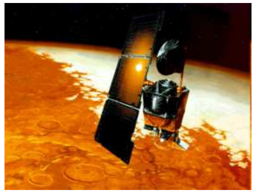
     ➔ relevant for the exam are all topics discussed during classes
       - if not explicitly stated otherwise

- ***any questions or „amendments“?***

In medias res…

- Why is software development so difficult?

FISCUS



Therac-25

TOLL COLLECT
service on the road

Baggage transport
Denver Airport

hochschulstart.de
Stiftung für Hochschulzulassung

Inpol-neu

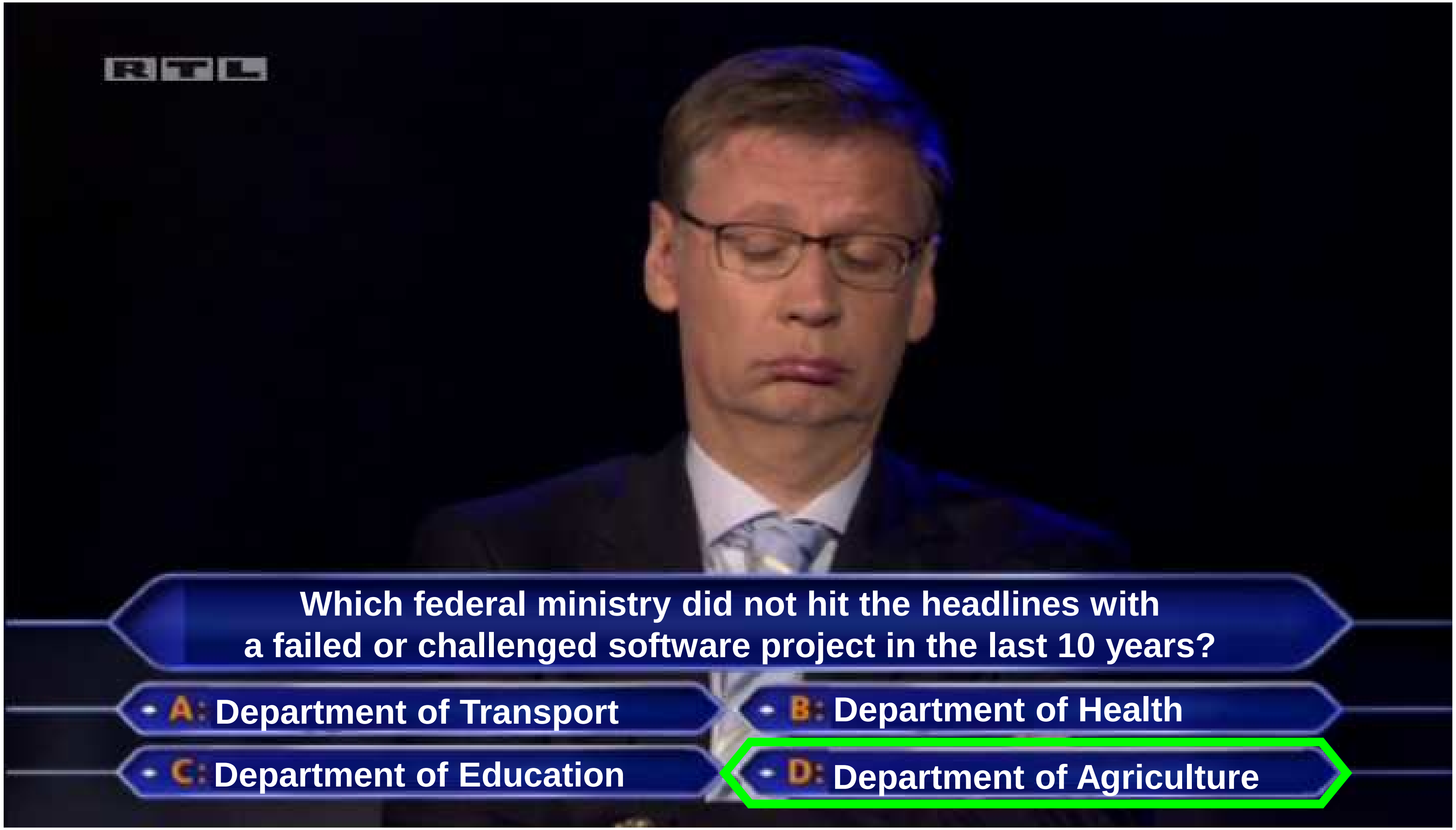
It is a very humbling experience to make a multi-million-dollar mistake, but it is also very memorable.

Turing Award Winner
Fred Brooks

SABRE

JOBBÖRSE

Who listened closely?



Which federal ministry did not hit the headlines with
a failed or challenged software project in the last 10 years?

A: Department of Transport

B: Department of Health

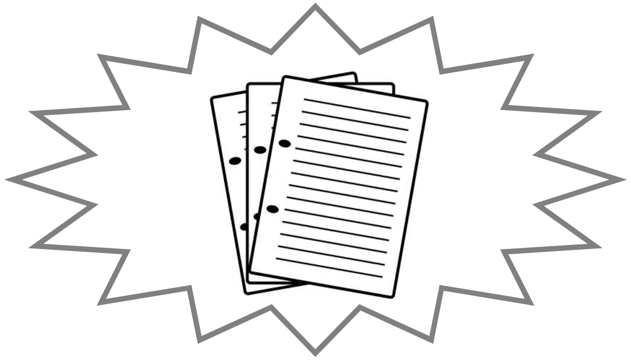
C: Department of Education

D: Department of Agriculture

Teaser I: A Kind of Magic?

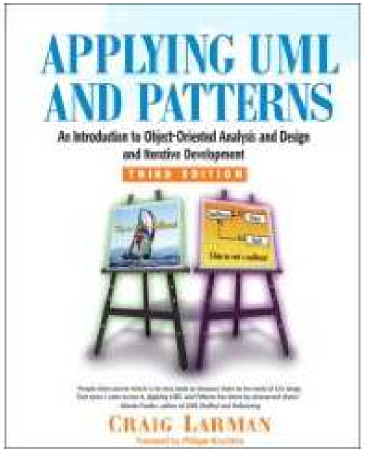
- How can we bridge the gap between requirements and code?
  - in a systematic manner?

What the customer really needed



Requirements

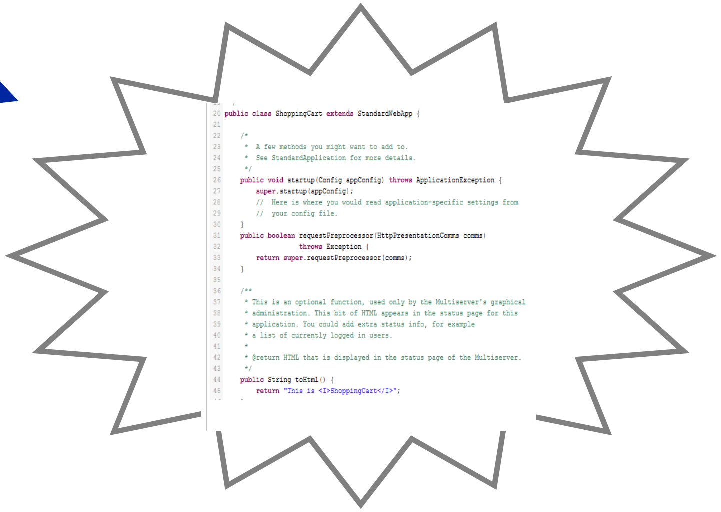
???


APPLYING UML AND PATTERNS

An Introduction to Object-Oriented Analysis and Design and Iterative Development

THIRD EDITION

CRAIG LARMAN



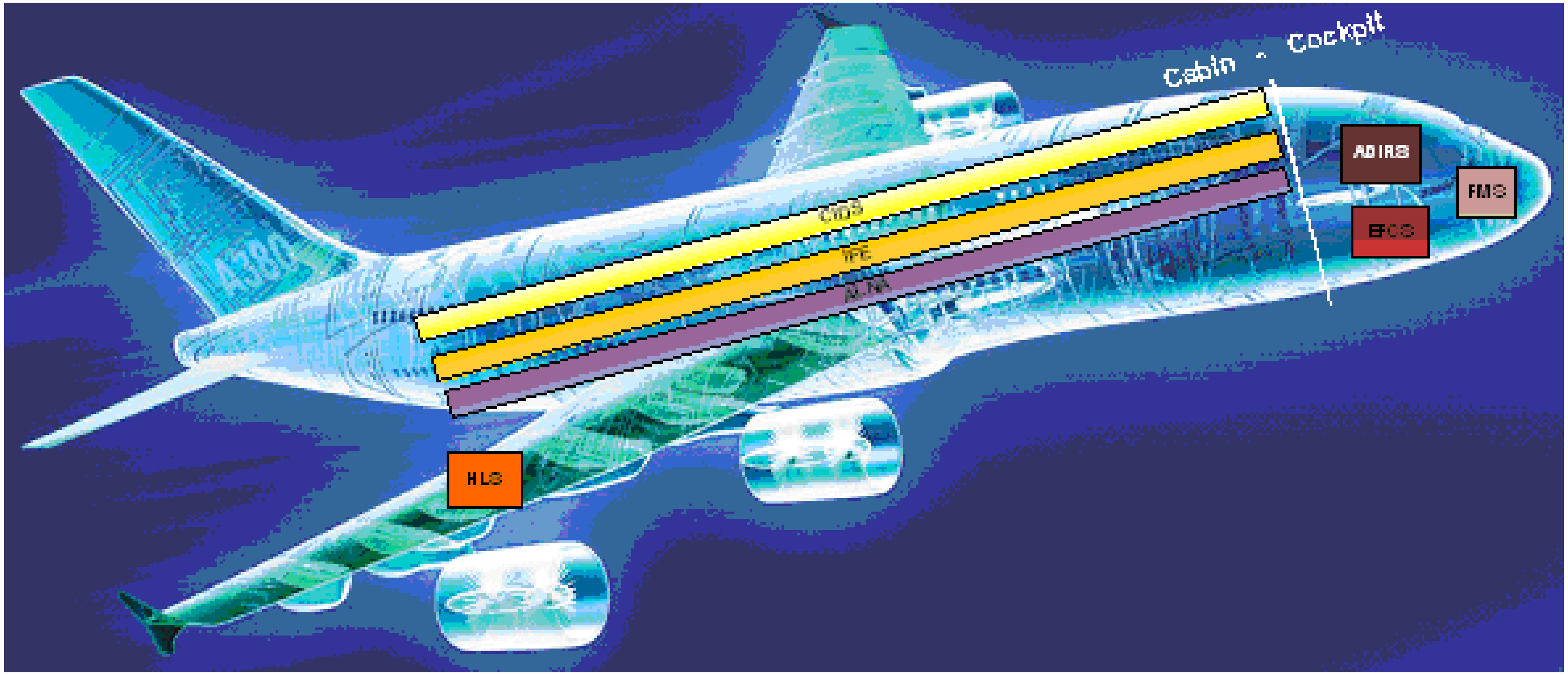
```
public class ShoppingCart extends StandardWebApp {

    /*
     *  A few methods you might want to add to.
     *  See StandardApplication for more details.
     */
    public void startup(Config appConfig) throws ApplicationException {
        super.startup(appConfig);
        //  Here is where you would read application-specific settings from
        //  your config file.
    }
    public boolean requestPreprocessor(HttpPresentationComms comms)
                   throws Exception {
        return super.requestPreprocessor(comms);
    }

    /**
     * This is an optional function, used only by the Multiserver's graphical
     * administration. This bit of HTML appears in the status page for this
     * application. You could add extra status info, for example
     * a list of currently logged in users.
     *
     * @return HTML that is displayed in the status page of the Multiserver.
     */
    public String toHtml() {
        return "This is <I>ShoppingCart</I>";
```

Code

What does this mean in Practice?

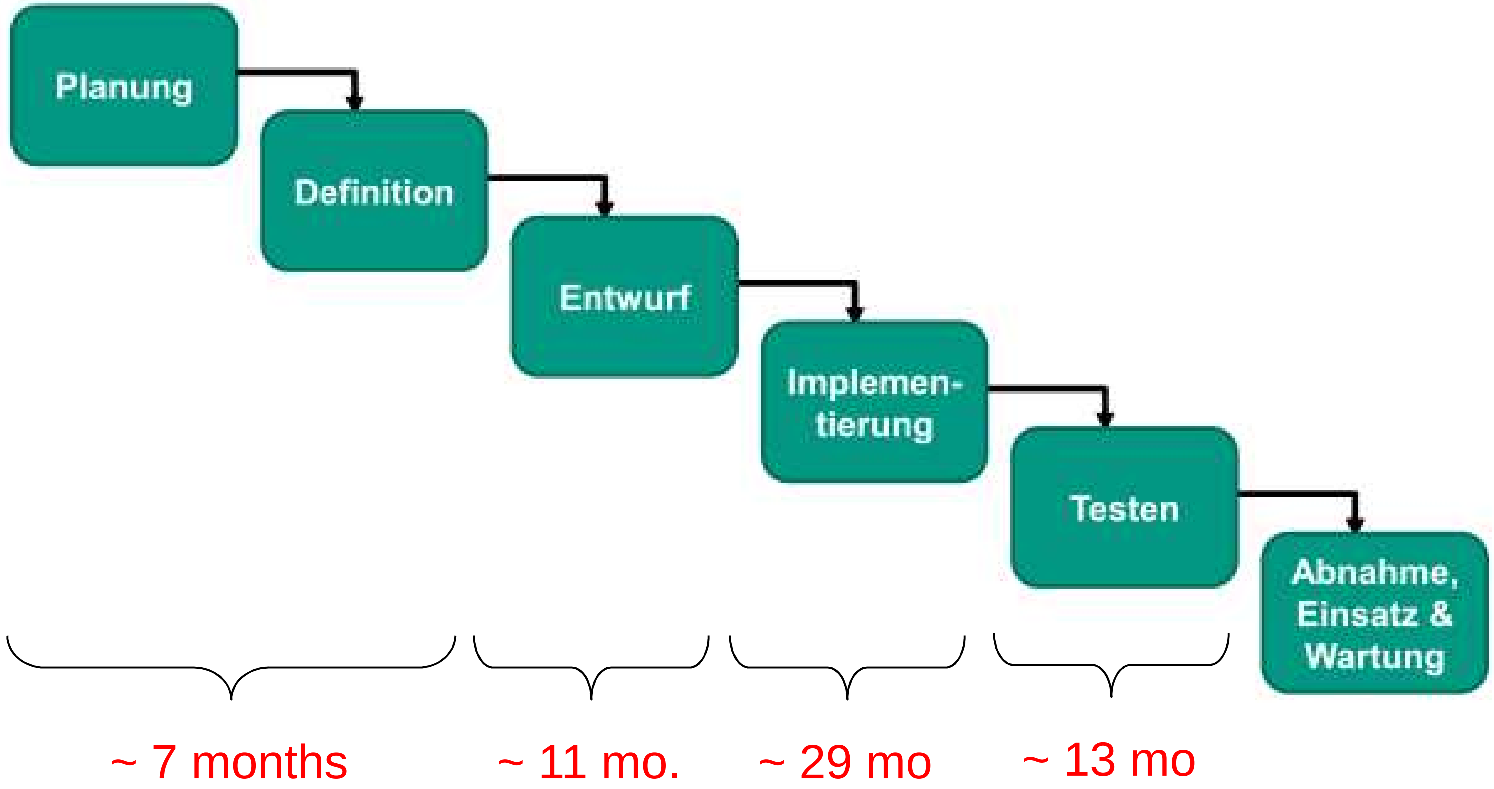


- Airbus 380
  - ~ 4 million parts
  - CIDS alone has more than 2 million LOC

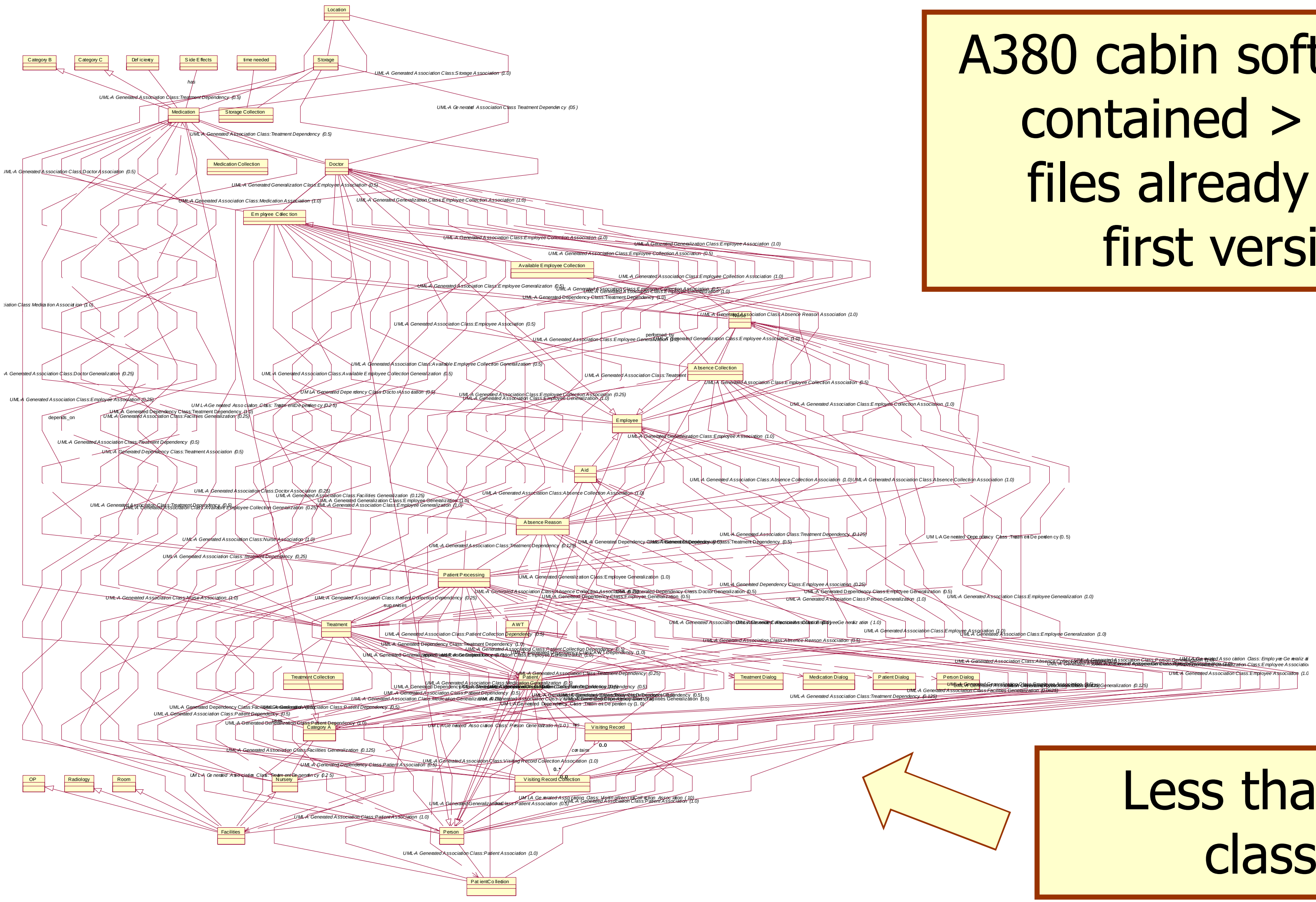
[Airbus]

Teaser II: „Remote“ Control

- Waterfall and schedule
  - take the initial version of A380 cabin software (~800 kLOC) as an example
  - ➔ estimated: ~ 4,500 person months and a schedule length of about 60 months *(average staff: 75 people)*



Teaser IIIb: Taming Complexity



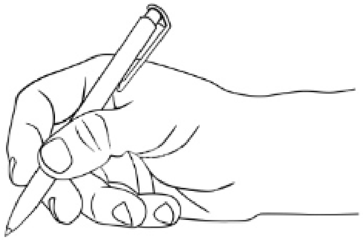
A380 cabin software contained > 4000 files already in its first version …

Less than 50 classes!!!

Further Teasers

1. Putting more people on a late project makes it _________________
2. Errors are more frequent during _____________________ phases
3. Testing can only show the presence of errors, but not their _________
4. A system that is used will be ____________
5. Only what is ___________ can be changed without risk

➔ Implications?



Coarse Course Schedule (1)

- Classes
  - on Mondays, 15:45 Uhr, Daimler Hörsaal
  - on Tuesdays, 9:45 Uhr, Hertz Hörsaal

Date	Tentative Content
Mo. 21.10.	Today: Warm-Up
Di. 22.10.	Software Processes
Mo. 28.10.	cont.
Di. 29.10.	Agile Development
Mo. 04.11.	*Guest Lecture by Andrena Objects*
Di. 05.11.	Requirements Elicitation
Mo. 11.11.	cont. + Use Cases
Di. 12.11.	cont.
Mo. 18.11.	Requirements Analysis
Di. 19.11.	cont.
Mo. 25.11.	Software Architecture
Di. 26.11.	cont. + Component-Based Architectures
Mo. 02.12.	cont.
Di. 03.12.	Persistence Patterns

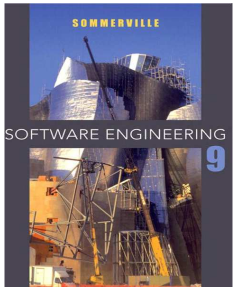
Coarse Course Schedule (2)

Date	Tentative Topic
Mo. 09.12.	Web Application Basics
Di. 10.12.	Software Security
Mo. 16.12.	UI Design
Di. 17.12.	Usability
Di. 07.01.2014	Software Design
Mo. 13.01.2014	cont. + Clean Code
Di. 14.01.	cont.
Mo. 20.01.	Real Time Patterns
Di. 21.01.	cont.
Mo. 27.01.	Model-Driven Software Development
Di. 28.01.	cont.
Mo. 03.02.	Software Reliability
Di. 04.02.	Project Management
Mo. 10.02.	Cost Estimation
Di. 11.02.	cont.
Mo. 04.02.	Buffer (for Left-Over Topics)
Di. 05.02.	Wrap-Up

➔ ***Exam: Friday, 14.03.13***

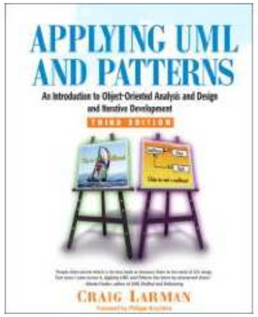
General Literature References

- Recommended materials
  - **General SE**
    - Ian Sommerville
      Software Engineering
      9th edition, Addison-Wesley, 2010



    - IEEE et al.
      Software Engineering Body of Knowledge (SWEBOK)
      - http://www2.computer.org/portal/web/swebok

  - **Agile Modeling**
    - Craig Larman
      Applying UML and Patterns
      3rd edition, Prentice Hall, 2004


APPLYING UML AND PATTERNS
An Introduction to Object-Oriented Analysis and Design and Iterative Development
THIRD EDITION
CRAIG LARMAN

➔ *Further references will be listed at the end of each lecture*

Geheimtipp für C# →
Geirhos, Galileo

Please refresh …

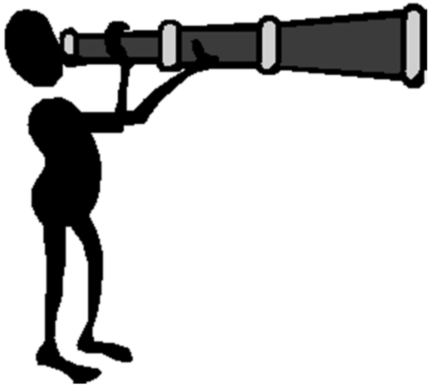
- … your knowledge of –
  - systematic software development
    - what models can be used for what purpose?
  - software processes, especially
    - the waterfall and V model

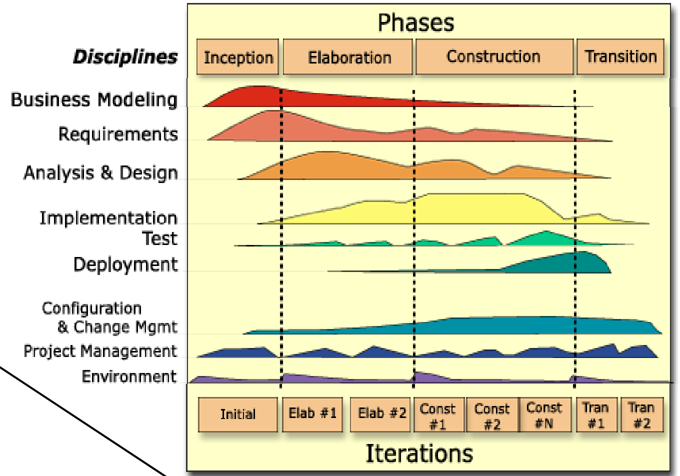
for tomorrow (or now).

Conclusion

- **Software development is a challenging activity**
  - still no routine

➔ A variety of novel approaches have been developed to overcome this challenge
  - *state of the art will be presented in this course*





- ***Thank you for your attention!***

Appendix: Some A380 Fun Facts [Airbus]

- A380 is the largest civil aircraft in history
  - with a maximum take-off weight of 123.5klb (560 tonnes)
- A380 was launched in December 2000
  - 40 years after 747
- A380 consumes less than 3 litres per pax per 100 km
  - 747-400 which consumes 3.4 litres per pax per 100km
- 6.000 engineers and technicians have worked on the program
- 10.000 bolts are inserted inside the fuselage to attach each of the 3 main parts
  - plus 8.000 to attach both wings
- Each A380 consists of around 4 million individual components
  - having 2.5 million part numbers
  - produced by 1500 companies from 30 countries around the world
- The weight of the external paint of the A380 is 531 kg