Foto: Thomas Josek

Universität zu Köln

Software Engineering

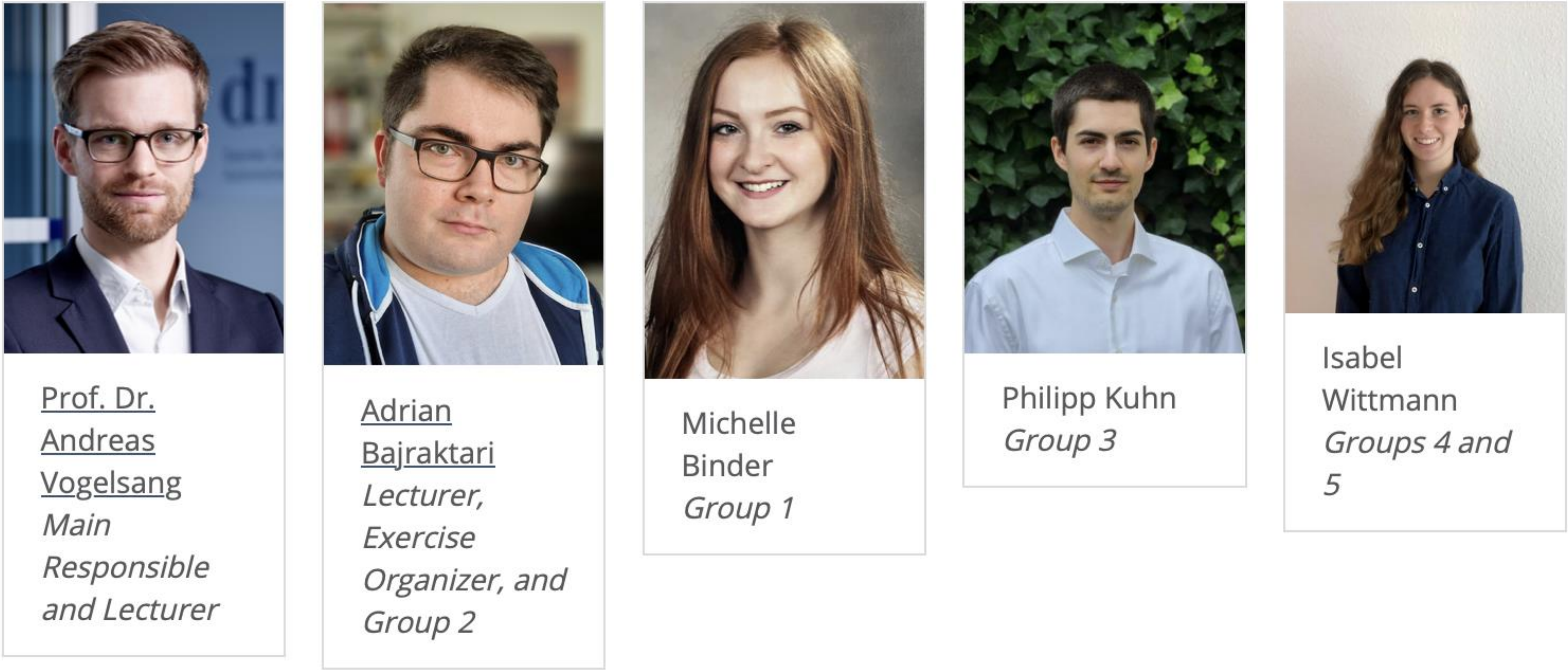
Introduction

Software & Systems Engineering | Prof. Dr. Andreas Vogelsang | 09.10.2023

@andivogelsang

✉ vogelsang@cs.uni-koeln.de

The Team



Prof. Dr. Andreas Vogelsang
Main Responsible and Lecturer

Adrian Bajraktari
Lecturer, Exercise Organizer, and Group 2

Michelle Binder
Group 1

Philipp Kuhn
Group 3

Isabel Wittmann
Groups 4 and 5

https://cs.uni-koeln.de/sse

Software & System Engineering @ UzK

Requirements Engineering

Automate RE as far as possible

- Requirements Classification
- Trace Link Recovery
- Glossary Term Extraction
- Detection of Quality issues in NL Requirements
- Test Case Creation from NL Requirements
- Kano categorization

Explainable Intelligent Systems

Explain intelligent and autonomous behavior to users

- Exp. System Requirements
- Exp. Modeling
- SE of Explainable systems
- Explainability and Ethics
- Development of (exp.) cyber-physical, IoT and autonomous systems in our SmartLab

Model-based Systems Engineering

Improve MBSE adoption in practice

- SPES Modeling Framework
- Feature Dependency Analysis
- Drivers and Hurdles of MBSE Adoption
- MBSE Adoption Strategies and Best Practices
- MBSE Maturity Model

Data-Driven Systems Engineering

Improve SE process by collecting and analyzing runtime data

- Integrate NL Requirements with System Executions
- Analyze Feature Usage based on Interaction Data

SE & ML

Integrate SE and ML techniques

- Requirements Engineering for ML Applications

Research Software Engineering

Improve Research Software Development using SE practices

- Code Quality and Principles
- Process and Team
- Validity and Reproducibility
- Credibility and Acknowledgement

Teaching in Software & Systems Engineering

Summer term

- Lecture Requirements Engineering (9 ECTS, Master)
- Lecture Empirical Software Engineering (6 ECTS, Master)

Winter term

- Lecture Software Engineering (9 ECTS, Bachelor)
- Capstone Project / Advanced SE Projects (Bachelor/Master)

Anytime

- Bachelor’s and Master's Thesis

DISCORD

- Discord will be our central instrument for announcements, questions, discussions, communication

- You are more than welcome to
  - Create new channels (public or closed)
  - Comment on and add to anything
  - Raise organizational questions directly to team members or post in channel **#orga-questions**
  - Raise questions on lecture content in channel **#lecture-questions**

- Use of real name is mandatory

Lecture: Software Engineering

- Credits: 9 ECTS (90 + 180h)
- Semester hours: 4+2
- Prerequisites (not formally)
  - Programmierkurs
  - Info I (Datastructures and Algorithms)
- Courses of Studies
  - B.Sc. Wirtschaftsinformatik, Physik
  - M.Sc. Mathematik, Wirtschaftsmathematik

- Exercise Groups
  - Group 1: Mondays, 12:00 - 13:30 (Michelle)
  - Group 2: Mondays, 14:00 - 15:30 (Adrian)
  - Group 3: Tuesday, 16:00 - 17:30 (Philipp)
  - Group 4: Wednesday, 10:00 - 11:30 (Isabel)
  - Group 5: Wednesday, 12:00 - 13:30 (Isabel)

Exercises: Software Engineering

- Exercise Groups
  - Group 1: Mondays, 12:00 - 13:30 (Michelle)
  - Group 2: Mondays, 14:00 - 15:30 (Adrian)
  - Group 3: Tuesday, 16:00 - 17:30 (Philipp)
  - Group 4: Wednesday, 10:00 - 11:30 (Isabel)
  - Group 5: Wednesday, 12:00 - 13:30 (Isabel)

- Agenda
  - Common mistakes and questions from last homework.
  - Repetition / few new content if necessary.
  - On-site exercises to practice for homeworks. Single and group work.
  - Outlook on next homework.
  - **Please always bring your laptop!**

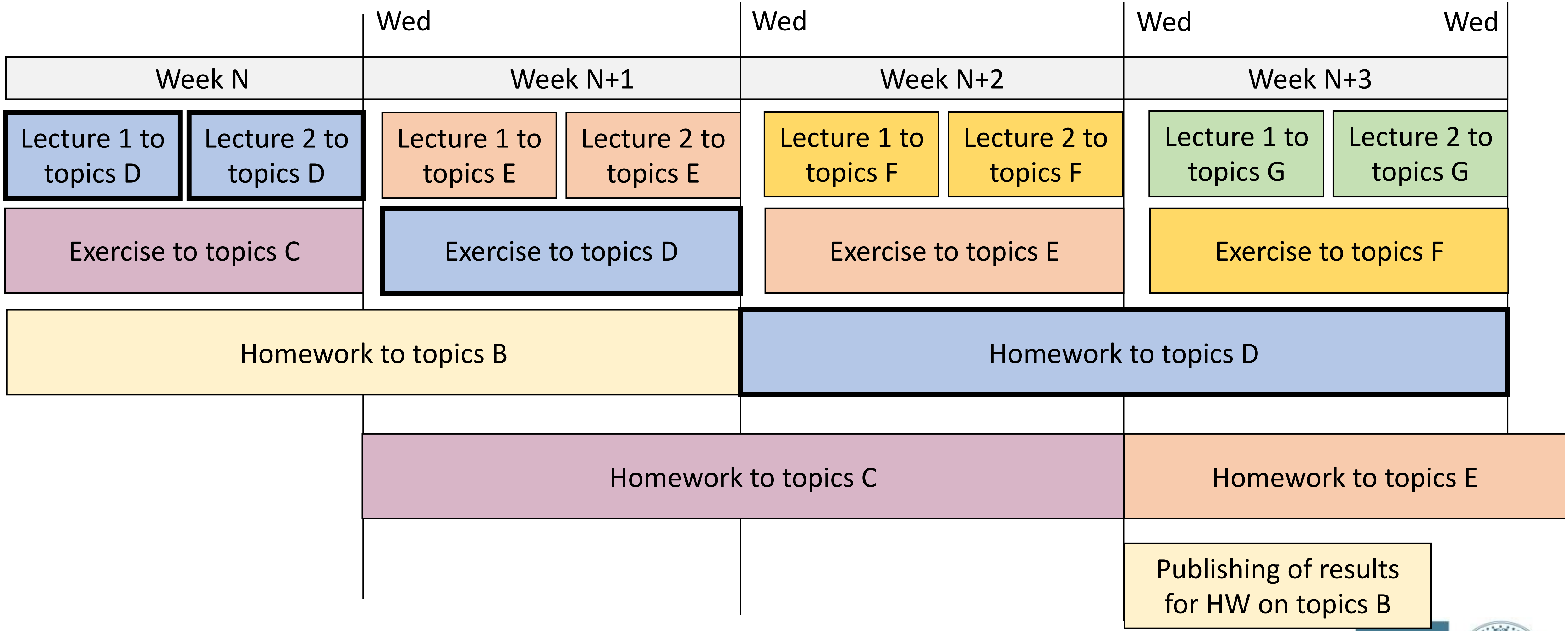
- Distribution of exercise groups via Ilias
- **Registrations starts today at 18:00**
  - First-Come, First-Serve

Homework

- Homework sheets will be published every Wednesday evening
- Homework sheets must be submitted two weeks later Wednesday at 16:00.
- Homework sheets will be graded.
- Based on the homework grading, you get
  - 5% bonus in the exam if you have >= 75% of points
  - 10% bonus in the exam if you have >= 90% of points

- Homework sheets can be submitted in groups of up to 3 students
  - All members of a group will get the same number of points (i.e., everyone is responsible for the results of the whole sheet)

Workflow



Wed

Wed

Wed

Wed

Week N

Week N+1

Week N+2

Week N+3

Lecture 1 to topics D

Lecture 2 to topics D

Lecture 1 to topics E

Lecture 2 to topics E

Lecture 1 to topics F

Lecture 2 to topics F

Lecture 1 to topics G

Lecture 2 to topics G

Exercise to topics C

Exercise to topics D

Exercise to topics E

Exercise to topics F

Homework to topics B

Homework to topics D

Homework to topics C

Homework to topics E

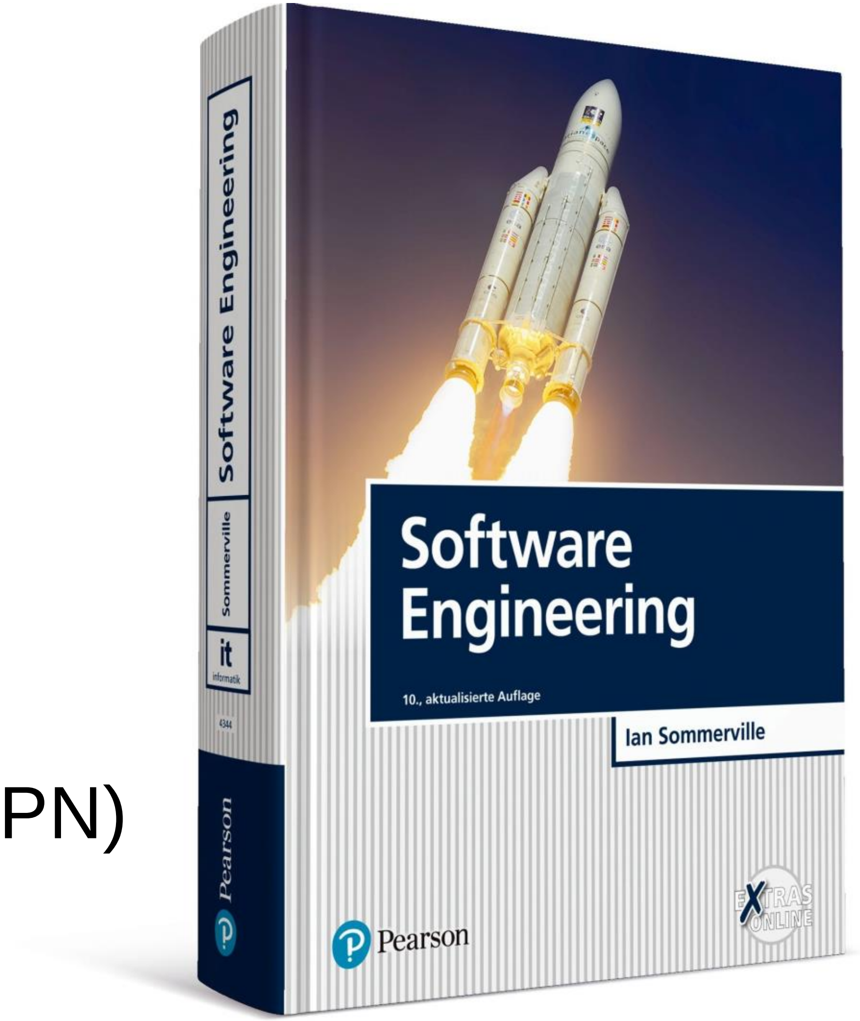
Publishing of results for HW on topics B

Exam

- 2 h written exam
  - First exam date: Feb 05, 12:30
  - Second exam date: Mar 27, 12:30
  - You need >50% points in the exam to pass
  - The bonus will only be counted if you pass the exam.
- The exam will be provided in German;
  answers can be given in German or English
- You are allowed to take 1 page of handwritten notes with you

Textbook

- There is no specific textbook for the lecture
- Recommended books
  - Sommerville, Software Engineering
  - Sommerville, Modernes Software Engineering
- Available online in the Info/Winfo Bib (via VPN)



Acknowledgements

The content and slides are largely influenced and partially reused by lectures of respected colleagues

Thomas Thüm, Universität Ulm

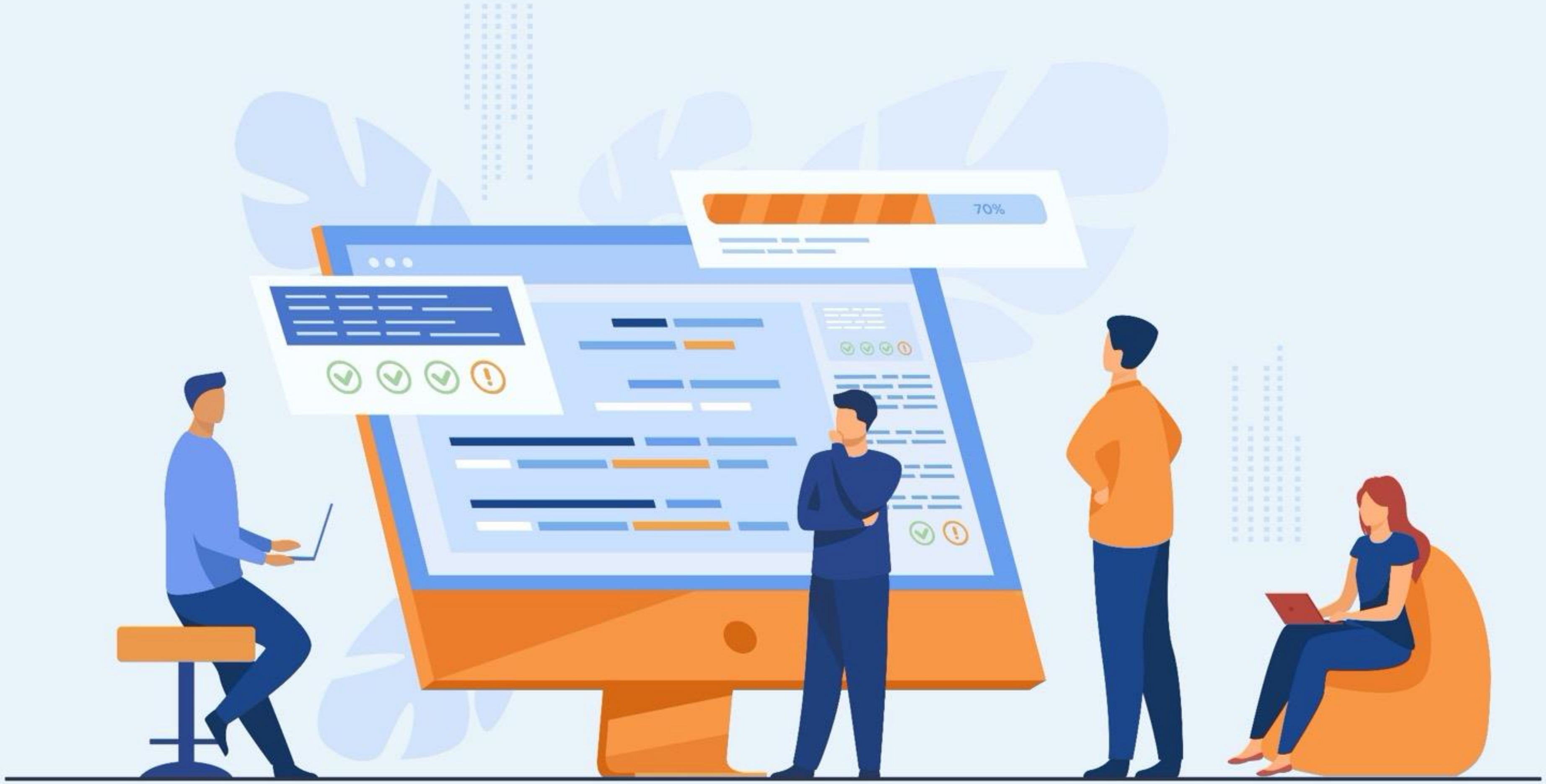
Software Engineering I + II

Christian Kästner, CMU

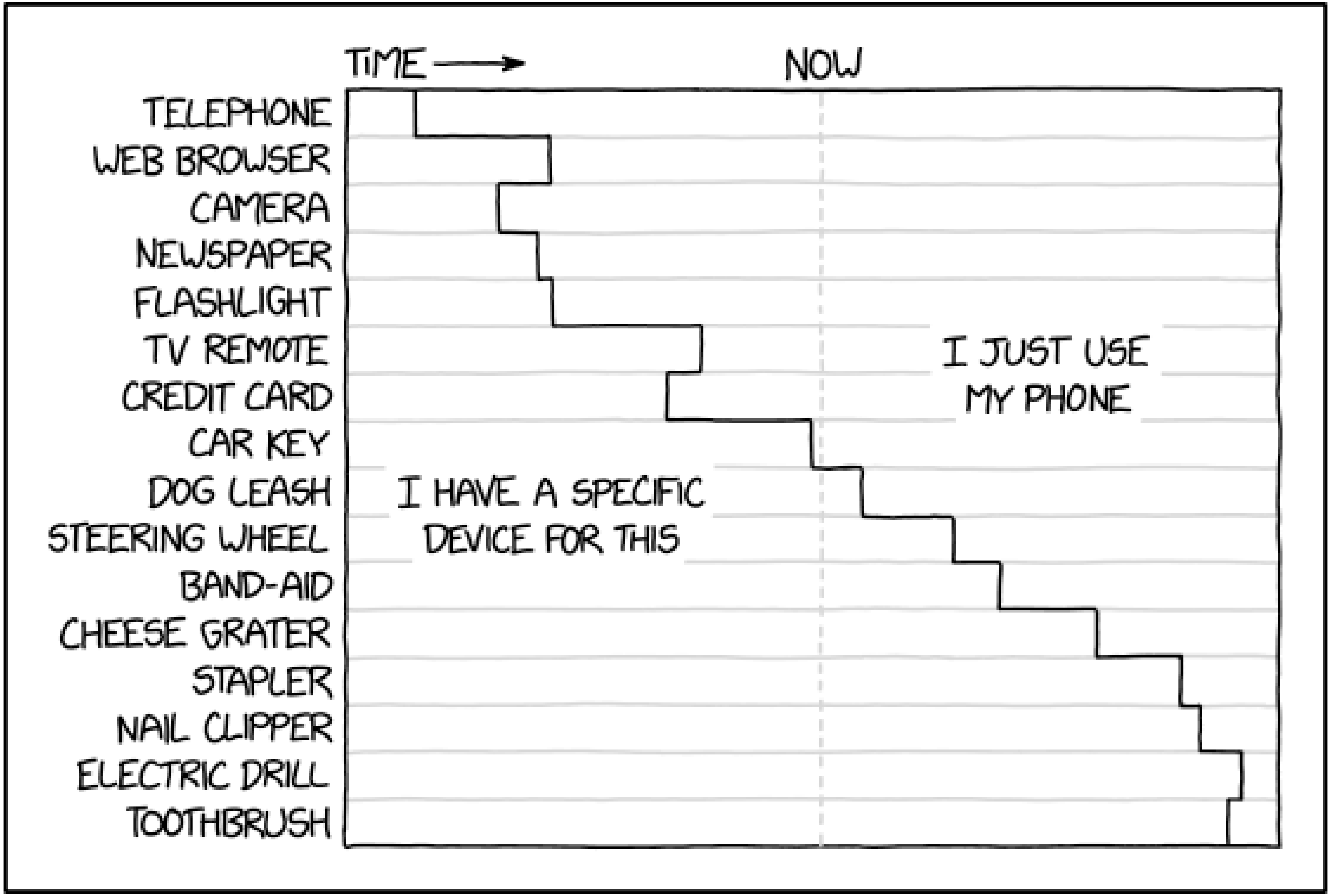
Foundations of SW Engineering

Jürgen Cito, TU Wien

Web Engineering



What is Software?

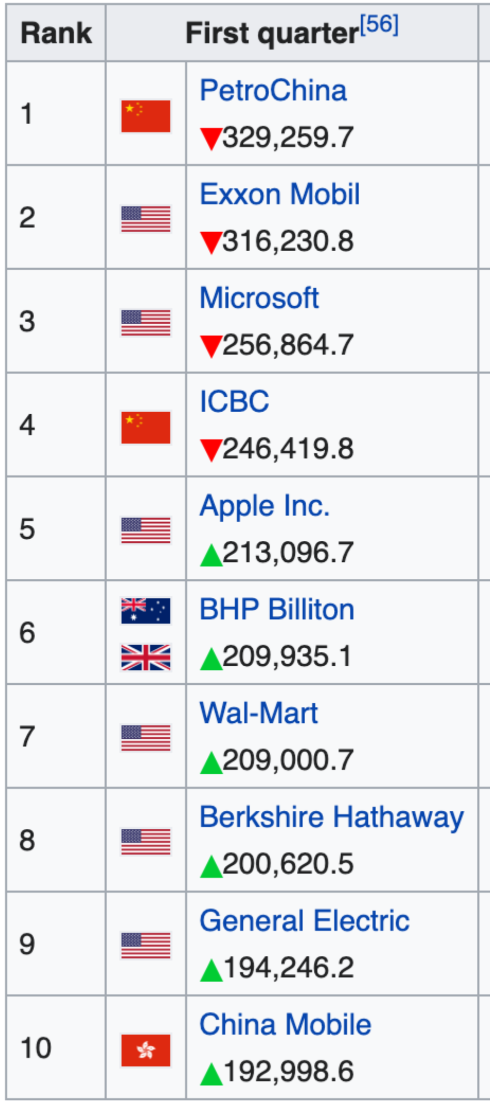

TIME →
NOW
TELEPHONE
WEB BROWSER
CAMERA
NEWSPAPER
FLASHLIGHT
TV REMOTE
CREDIT CARD
CAR KEY
DOG LEASH
STEERING WHEEL
BAND-AID
CHEESE GRATER
STAPLER
NAIL CLIPPER
ELECTRIC DRILL
TOOTHBRUSH
I HAVE A SPECIFIC DEVICE FOR THIS
I JUST USE MY PHONE

Software is eating the world.

Marc Andreessen

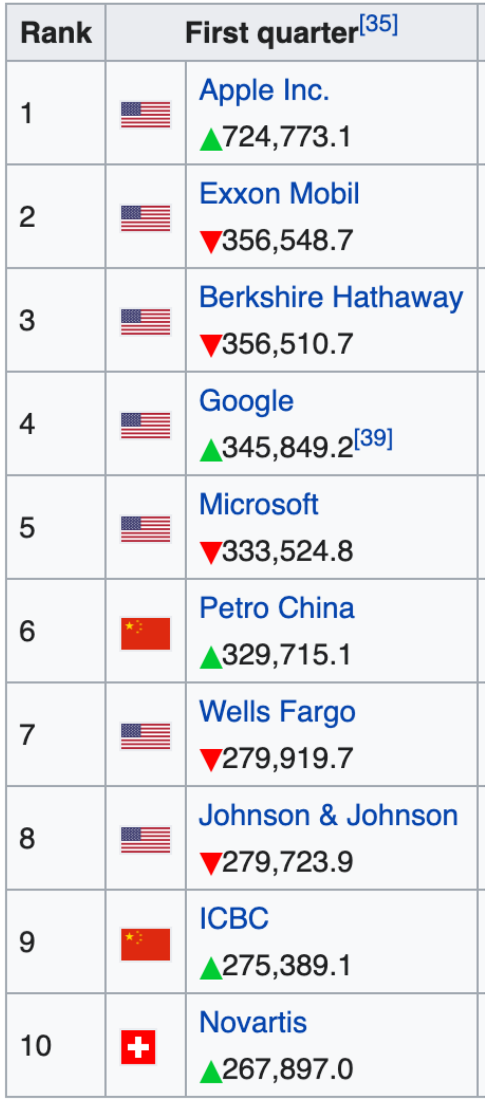
Publicly traded Companies by Market Capitalization

2010



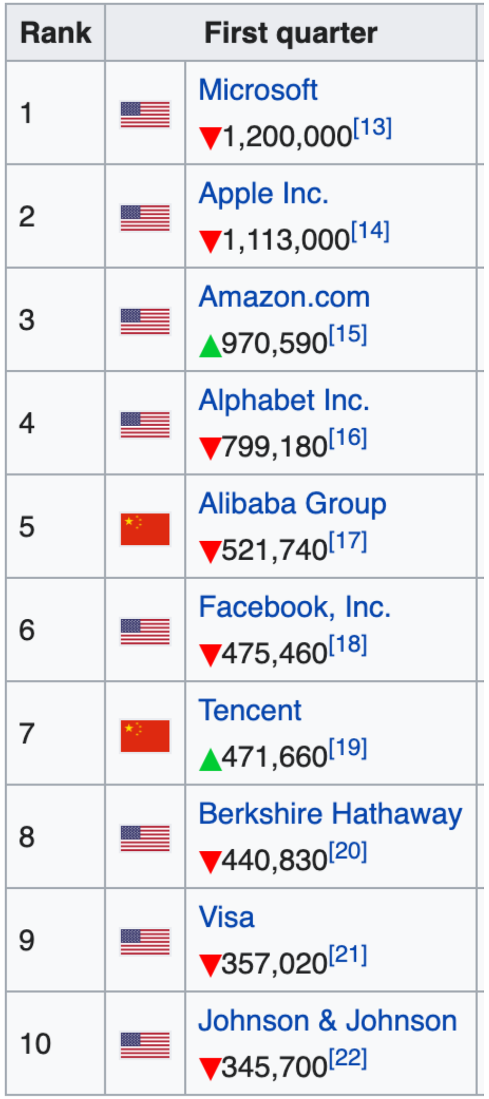
Rank	First quarter[56]	
1	🇨🇳	PetroChina ▼329,259.7
2	🇺🇸	Exxon Mobil ▼316,230.8
3	🇺🇸	Microsoft ▼256,864.7
4	🇨🇳	ICBC ▼246,419.8
5	🇺🇸	Apple Inc. ▲213,096.7
6	🇦🇺 🇬🇧	BHP Billiton ▲209,935.1
7	🇺🇸	Wal-Mart ▲209,000.7
8	🇺🇸	Berkshire Hathaway ▲200,620.5
9	🇺🇸	General Electric ▲194,246.2
10	🇭🇰	China Mobile ▲192,998.6

2015



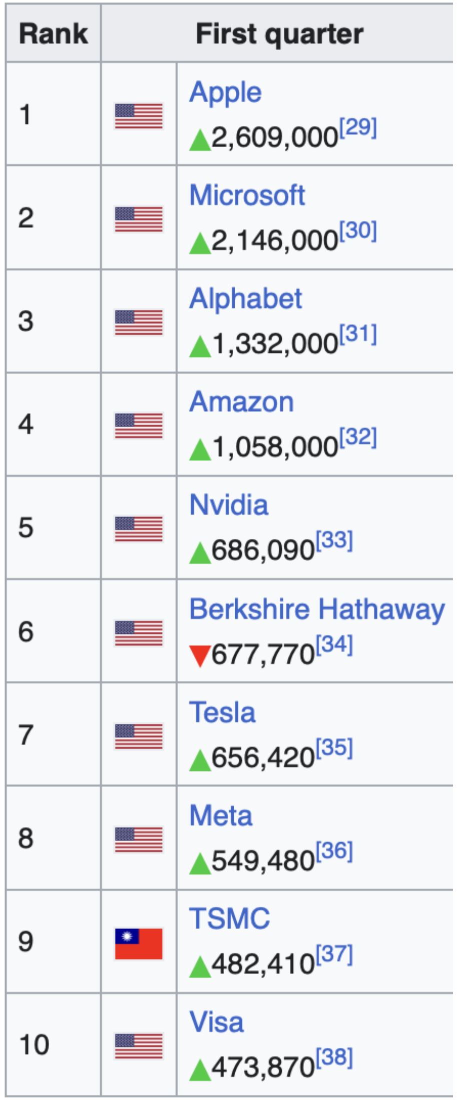
Rank	First quarter[35]	
1	🇺🇸	Apple Inc. ▲724,773.1
2	🇺🇸	Exxon Mobil ▼356,548.7
3	🇺🇸	Berkshire Hathaway ▼356,510.7
4	🇺🇸	Google ▲345,849.2[39]
5	🇺🇸	Microsoft ▼333,524.8
6	🇨🇳	Petro China ▲329,715.1
7	🇺🇸	Wells Fargo ▼279,919.7
8	🇺🇸	Johnson & Johnson ▼279,723.9
9	🇨🇳	ICBC ▲275,389.1
10	🇨🇭	Novartis ▲267,897.0

2020



Rank	First quarter	
1	🇺🇸	Microsoft ▼1,200,000[13]
2	🇺🇸	Apple Inc. ▼1,113,000[14]
3	🇺🇸	Amazon.com ▲970,590[15]
4	🇺🇸	Alphabet Inc. ▼799,180[16]
5	🇨🇳	Alibaba Group ▼521,740[17]
6	🇺🇸	Facebook, Inc. ▼475,460[18]
7	🇨🇳	Tencent ▲471,660[19]
8	🇺🇸	Berkshire Hathaway ▼440,830[20]
9	🇺🇸	Visa ▼357,020[21]
10	🇺🇸	Johnson & Johnson ▼345,700[22]

2023



Rank	First quarter	
1	🇺🇸	Apple ▲2,609,000[29]
2	🇺🇸	Microsoft ▲2,146,000[30]
3	🇺🇸	Alphabet ▲1,332,000[31]
4	🇺🇸	Amazon ▲1,058,000[32]
5	🇺🇸	Nvidia ▲686,090[33]
6	🇺🇸	Berkshire Hathaway ▼677,770[34]
7	🇺🇸	Tesla ▲656,420[35]
8	🇺🇸	Meta ▲549,480[36]
9	🇹🇼	TSMC ▲482,410[37]
10	🇺🇸	Visa ▲473,870[38]

Software Characteristics

Abstract (no physical laws)

intangible
(„immatriell“)

Hard to measure
(“Technical data“ of Software?)

Frequent adaptation

Easier to adapt than HW

No deterioration

aging

No spare parts

Software

Software

Software stands for one or several computer programs and all associated documentation, libraries, support websites, and configuration data that are needed to make these programs useful.

[adapted from Sommerville]

Explanation

The term program is used in a broader sense here. Software may also include source code, software models, or binaries.

Software Products

Software Product and Professional SW

A software product is a software that can be sold to a customer. Professional software is software intended for use by someone apart from its developer and it is usually developed by teams rather than individuals.

[adapted from Sommerville]

Application SW and System SW

Application SW or Application

Software that is designed for end users and applied for certain purposes. (“Anwendungssoftware” oder “Anwendung”)

Examples

web browsers, media players, email or chat clients, text or photo editors, games

System SW

Software that is not application software and typically being designed to provide a platform for other software.

Examples

operating systems, game engines, GUI frameworks

Classification not always unique

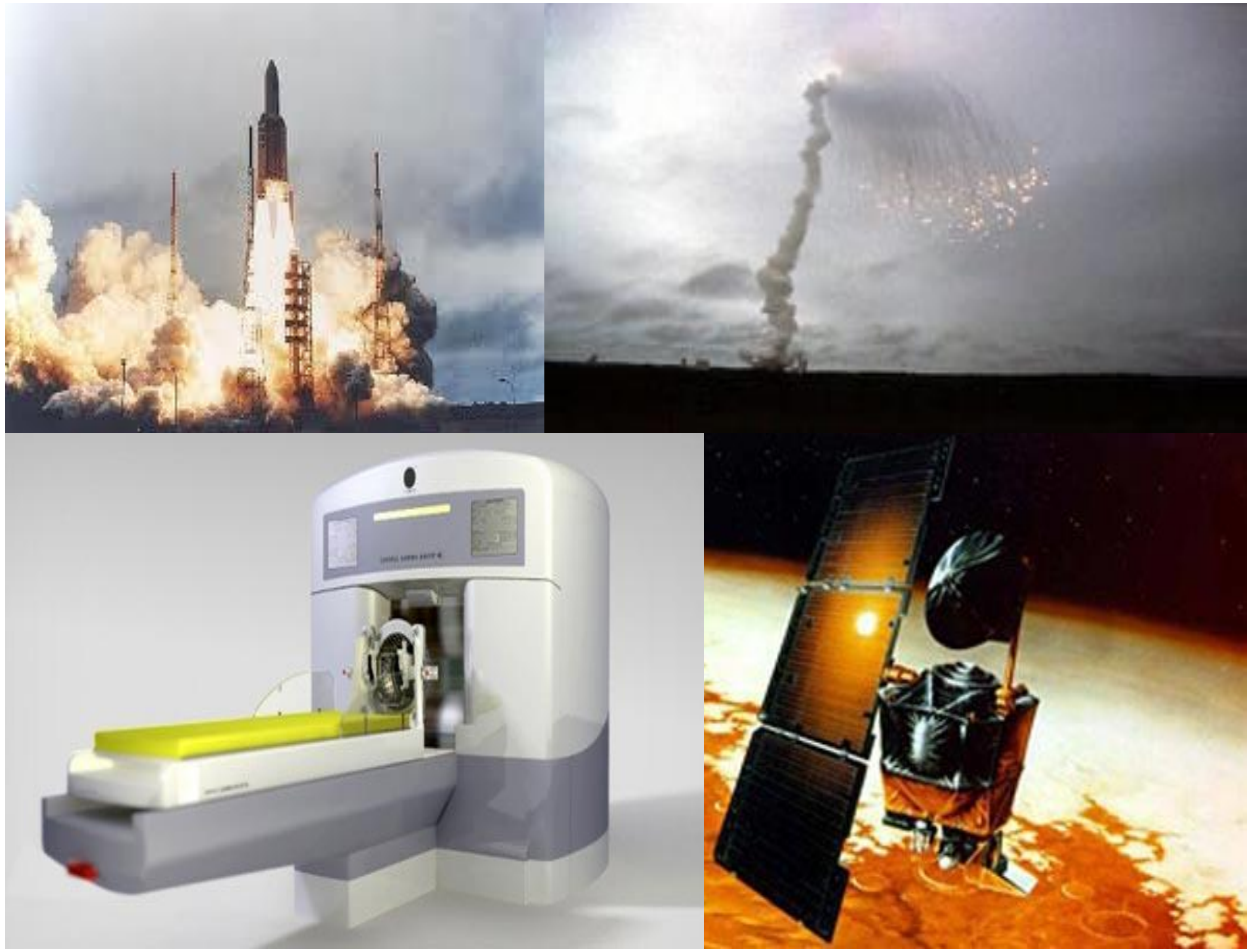
e.g., web browsers and chat clients take over more and more features of operating systems

Software-intensive Systems

Software-intensive Systems
(aka Cyber-Physical Systems)

A system (i.e., mechanics, electronics, software) where software influences to a large extent the design, construction, deployment, and evolution of the system as a whole.

Software Defects

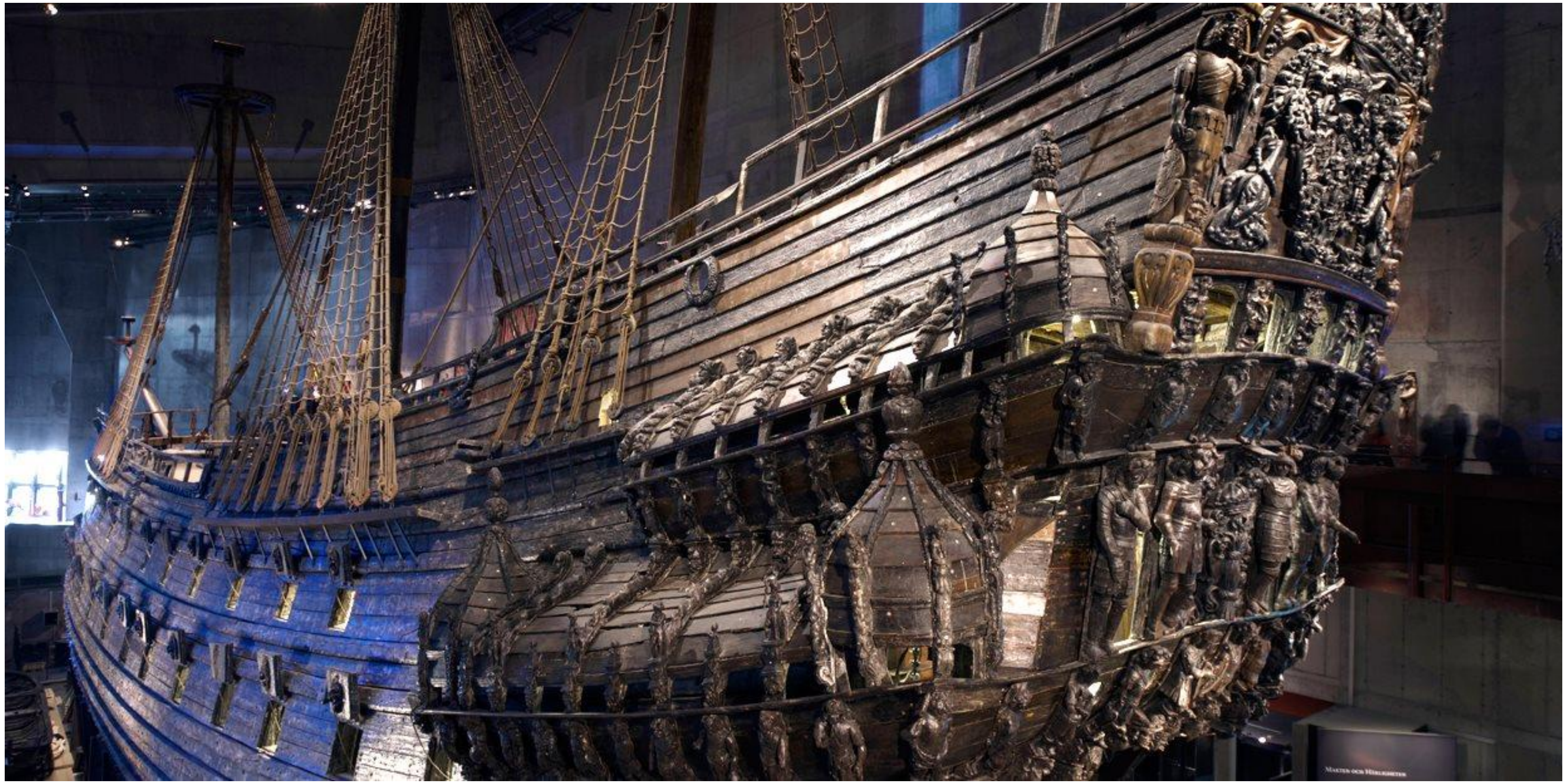


top-10-list.org/2010/05/03/ten-costliest-software-bugs/

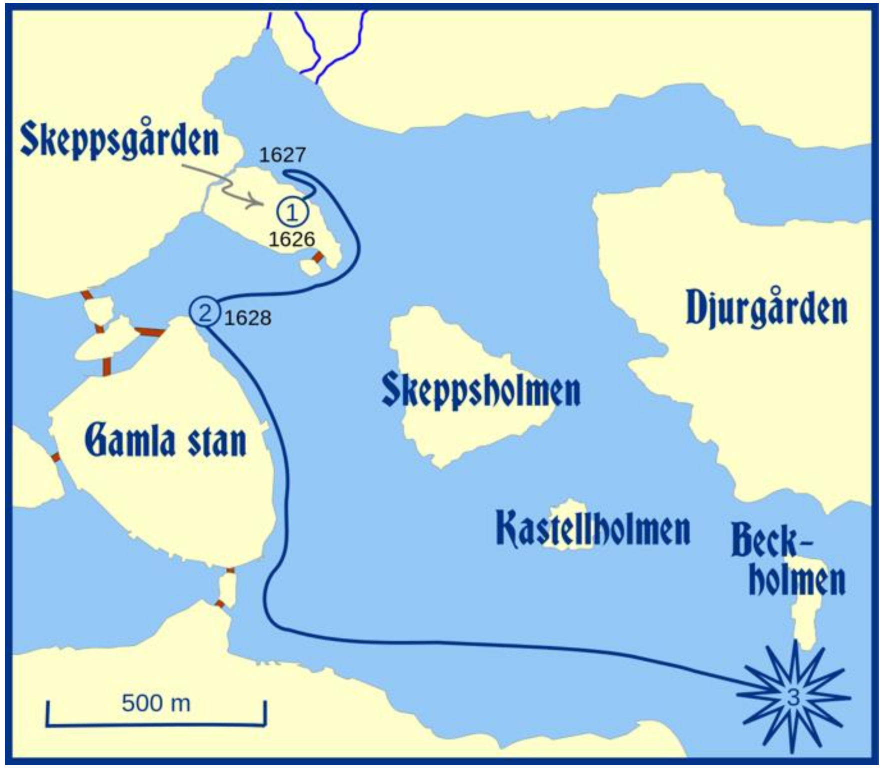
Reverse thrust must only be activated
if airplane is on the ground

The airplane is on the ground
if weight on each wheel is > 6t
AND all wheels turn with > 133 km/h

Vasa



Vasa


Skeppsgården
1627
1
1626
2
1628
Djurgården
Skeppsholmen
Gamla stan
Kastellholmen
Beck-
holmen
3
500 m

What happened?

- Changing shipbuilding orders
- No specification for modified keel
- Changing armaments requirements

Requirements Engineering

- Death of shipbuilder one year before completion

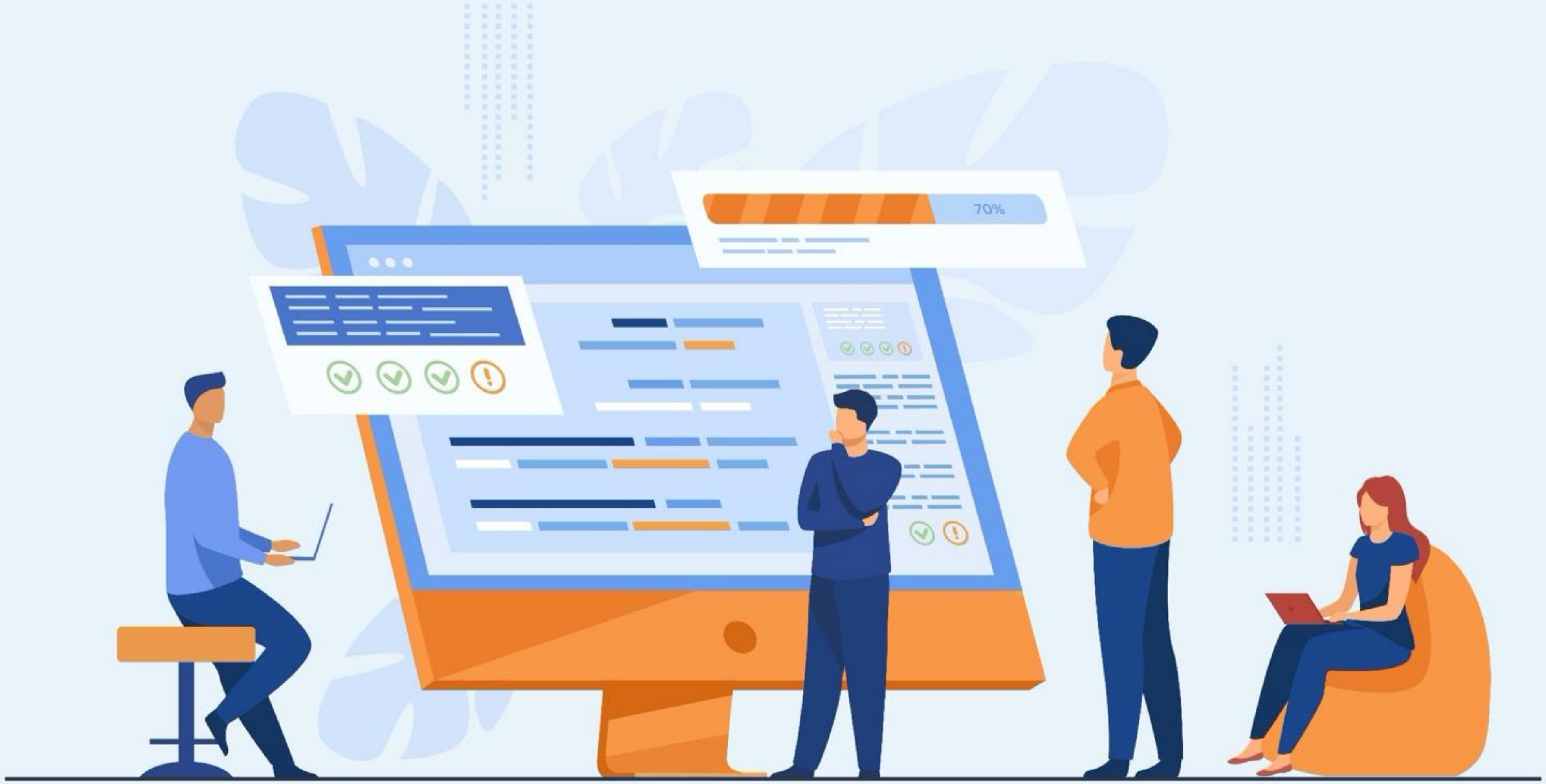
Team Dynamics

- No possibility to calculate stability, stiffness, or sailing characteristics

Measurability

- Failed pre-launch stability test

QA



What is Software Engineering?

Software Engineering vs. Programming

1968 NATO Conference on Software Engineering in Garmisch Partenkirchen

- Birth of the „Software Crisis“
  - Provocative title
  - Call for Action

Software Engineering

The establishment and use of sound engineering principles in order to obtain economically software that is reliable and runs on real machines.
(F.L. Bauer, NATO-Conference Software-Engineering 1968)

Margret Hamilton

- Main responsible for On-board flight SW in the NASA Apollo Program

- No education in Software or Systems Engineering

- Made the term „Software Engineering“ popular

1969

https://www.designnews.com/electronics-test/no-choice-be-pioneer-story-margaret-hamilton

More recent…

Software Engineering

Software engineering is an engineering discipline that is concerned with all aspects of software production from **initial conception** to **operation** and **maintenance**. [...] Software engineering is not just concerned with the **technical processes** of software development. It also includes activities such as **software project management** and the **development of tools, methods, and theories** to support software development.

[Sommerville]

(Software) Engineering

Engineering

Engineering is about getting results of the required **quality** within **schedule** and **budget**. [...] Engineers make things work. They apply theories, methods, and tools where these are appropriate. However, they use them **selectively** and always try to **discover solutions** to problems even when there are no applicable theories and methods. Engineers also recognize that they must work within **organizational** and **financial constraints**, and they must look for solutions within these constraints.

[Sommerville]

Software Engineering and Computer Science

SE vs. CS

Computer science focuses on theory and fundamentals; software engineering is concerned with the practicalities of developing and delivering useful software. [...] Computer science theory, however, is often most applicable to relatively small programs. Elegant theories of computer science are rarely relevant to large, complex problems that require a software solution.

[Sommerville]

Topics of the Lecture

- Requirements Engineering
- Modeling
- Architecture
- Advanced Programming
- Software Design & Implementation
- Web Engineering
- QA and Testing
- Deployment and Operations
- SE Processes, Software Management
- Software Economics and Ethics

What you will learn in this lecture

- How to write SW in a good way (instead of “hacking”)

- Will help you if you…
  - need to develop SW in teams (e.g., Capstone project)
  - need to develop SW with an impact (financial impact, safety impact,…)
  - need to manage a team of SW programmers

- Along the way, you will improve your programming and technical skills

Next steps

- Register to Discord
- Check course schedule and Ilias course

- Today (Monday) 18:00: Register for exercise group

- Wednesday: No lecture, instead
  - Git introduction video
  - Working sheet on technical setup (not graded but essential for all following sheets, to be done until your first exercise group)

- Monday 16.10.: Second lecture & exercise groups start

Universität zu Köln