TDDD89 - Scientific Method

Lecture 2

15:15 Thesis (project) types (continuation of Introduction lecture)

Ca. 15:45 Break

16:00 Panel discussion:

Outlook to working in international academic and industrial R&D contexts -
From the thesis project towards a professional career in industry or academia

Christoph Kessler

2024

LINKÖPINGS UNIVERSITET

Most common thesis types

Engineering vs. Science

	Engineering	Science
Rationale	Solve a problem	Gain understanding
Activities	Design, implement, verify	Interviews, experiments, proofs, ...
Goal	Satisfied customers	New/shared understanding

Often, no clear separation line in-between

Solid, trustworthy methodology and proper documentation is important for both →

Engineering vs. Science (cont.)

In order to **solve a problem**,
you need to **gain understanding** of the problem

In order to **verify** your implementation,
you may need to do **experiments, interviews, or proofs**

In order to have **satisfied customers**,
you need to achieve a **shared understanding** that the problem has been solved appropriately

Some common thesis types

- **Evaluations** of new techniques or methods to improve existing products or processes
- Design of a **prototype** application
- **Incremental** improvements of existing techniques of methods
- …

typical for **industry thesis projects** (in many cases even for academic projects)

Let us consider some examples …

1. Evaluation

Example:

General problem:
Does the code quality deteriorate over time? How do we know?

Approach 1:
Relate Git commits to code metrics such as *cyclomatic complexity* and draw a graph …

Why is this not a good idea?

1. Evaluation

Why is this not a good idea?

- We have not defined what we mean by ”code quality”, and hence, we have no way of knowing what to measure, or whether it relates to our desired quality.
- There is no clear sense of how to assess what we have done.
- There is no mention of how this would be useful to know.

1. Evaluation

General problem:
Does the code quality deteriorate over time? How do we know?

Approach 2:
Based on interviews, we define *code quality* as *detected faults*.
We determine whether detected faults *correlate* with cyclomatic complexity.
We define the *purpose* as being able to answer the question (RQ)
”What do we need to improve in order to produce long-term maintainable software?”

1. Evaluation

Why is this a better approach?

- We now have a definition of code quality
- The result can be assessed

2. Prototype design

Example:

General problem:
Create a new Foo application at our company

> **Approach 1:**
> Read about the latest techniques that can be used on Wikipedia and on project sites, implement the system and ask the company supervisor if he/she is happy

2. Prototype design

Why is this not a good approach?

- We don't know *why* the company wants the Foo application, how to evaluate it, or how long time it would take to implement it in full.
  - IF the requirements are not clear from the start, and the estimated time to implement the working, full solution is > 10 weeks, **do not aim for a full solution**

Side note: Especially, for external (company) thesis projects:
A proper thesis plan, with clearly formulated problem (*in your own words!*), requirements, priorities, risk analysis and schedule, gives better protection against possible misunderstandings and a "moving target" project

2. Prototype design

Approach 2:
If the projected time to implement a full solution is > 10 weeks

- Conduct a set of semi-structured interviews
  to understand the problem domain and the goal,
- a literature survey
  to understand solutions to similar problems,
- and a few structured iterations of prototype development and documented customer feedback,
  to produce *a set of requirements* based on the initial prototypes.

2. Prototype design

Approach 3:
If the projected time to implement a full solution is $\leq$ 10 weeks

- Determine functional and **non-functional requirements**,
- Do a literature survey
  to understand solutions to similar problems and **how to assess them**,
- Develop the application iteratively,
  and **evaluate the resulting application** based on the non-functional requirements

3. Incremental improvement

Example:

General problem:
We would like to perform testing of Telecom equipment with less hardware resources

Approach 1:
Implement a booking system that automatically releases resources upon expired time slots.

3. Incremental Improvement

Why is this not a good approach?

- We do not know how and why people use hardware resources, so we do not know how to optimize something.
- Is this a technical problem, an organizational problem or a cultural issue?
- How do we even *measure* utilization?

3. Incremental Improvement

Approach 2:

- Conduct an interview series to establish how different people perceive the problem
- Conduct an observational study to determine how people actually use resources
- Find a suitable model for resource utilization in the literature and apply it
- Measure utilization and relate to the results of the interviews

Summary

- The three most common types of final thesis in industry (and even academia):
  - evaluations,
  - prototypes, and
  - improvements.

- A **great thesis** is a marriage between solid engineering skills, genuine scientific approach to validate your work, and a lucid presentation.

What’s next?

Recall: Early deadlines

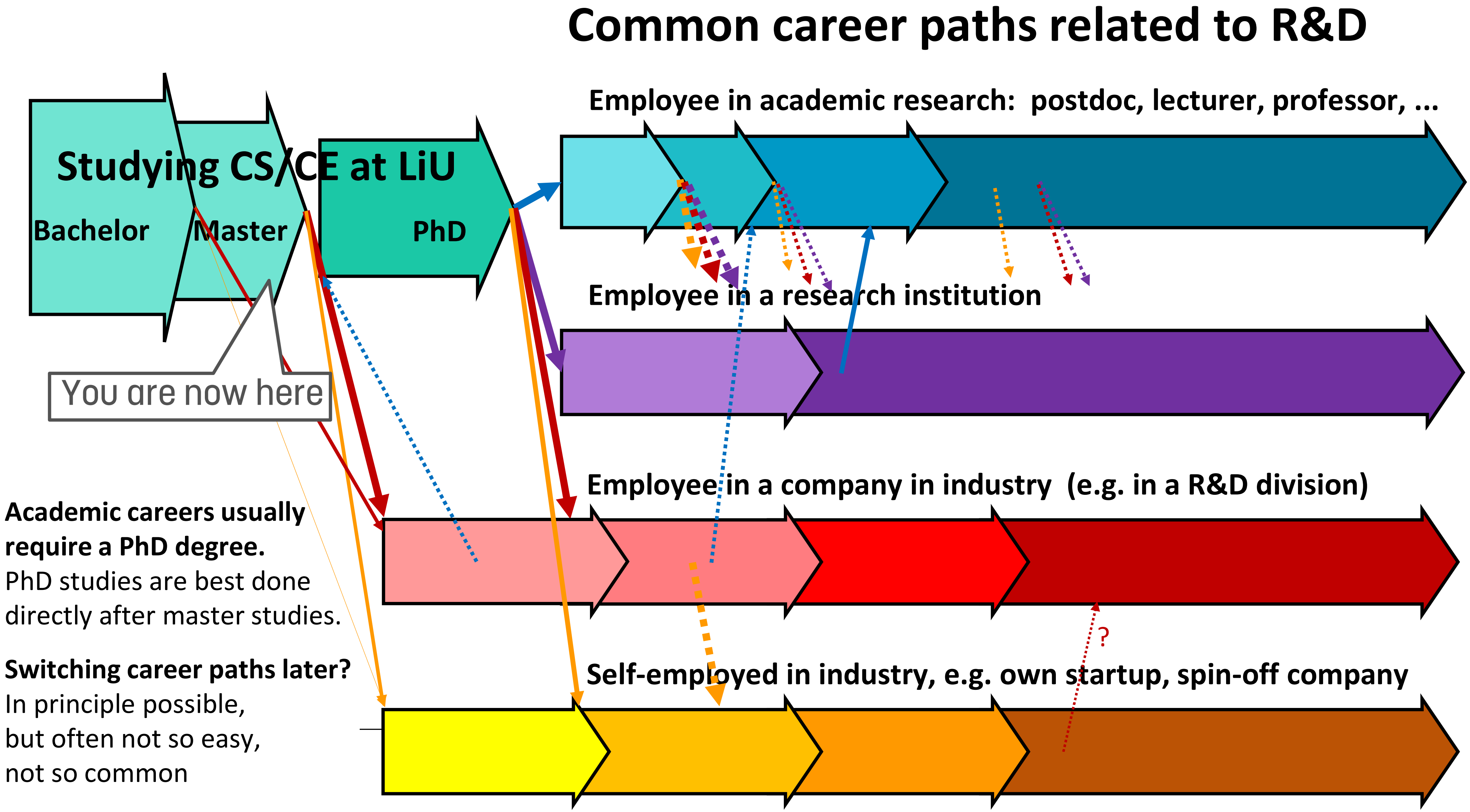
- **By Thursday**: Find a partner and a topic to work with in the seminars
  - Spreadsheet for matchmaking in Lisam cooperative area
- and submit a **thesis topic outline**
  - by storing a file (PDF) named "liuid001_liuid002.pdf" in LISAM cooperative area folder "Thesis_topic_outlines_by_7_Nov", containing:
    - Your and your seminar partner's **names and LiU-IDs**
    - Descriptive **title** of to-be-planned thesis project
    - **Abstract / short description** (e.g., copy in from company's proposal text) and keywords
    - **Closest topic area(s) and method types (1..6)** – the most relevant one first
- You will be divided into **seminar groups** in webreg based on your topic preferences.
  - To be announced on Monday **11/11** morning
- **Submissions for Seminar 1** (Lisam cooperative area, folder "Seminar 1") are due on **13/11**.
  - Start with the quiz and questions 6, 8-10, as these are not dependent on the sample thesis

Part 2:

Outlook to working in international academic and industrial R&D contexts –

From the thesis project towards a professional career in industry or academia

- **Short introduction**
- Break
- **16:00 Panel discussion with invited speakers**
  - relevant for Seminar 1 preparatory questions 8-10, due next wednesday


Common career paths related to R&D
Studying CS/CE at LiU
Bachelor
Master
PhD
You are now here
Employee in academic research: postdoc, lecturer, professor, ...
Employee in a research institution
Employee in a company in industry (e.g. in a R&D division)
Self-employed in industry, e.g. own startup, spin-off company
?
Academic careers usually require a PhD degree.
PhD studies are best done directly after master studies.
Switching career paths later?
In principle possible,
but often not so easy,
not so common

Some advice for planning the transition after master studies

- If you primarily target an academic career:
  - Good grades are a prerequisite
  - Do an *internal* master thesis project in the area you would like to specialize in, and do it *alone*
    - Talk to the corresponding professor(s) *early* and discuss suitable research-affine thesis topics
  - Plan to do the PhD studies early (ideally directly after master)

- If you primarily target an industry career:
  - Do finish your thesis and your degree *before* starting your first fulltime job
  - If you want to keep an option for later doing a PhD and possibly switching to an academic career, try to target positions in R&D

After the break, starting at 16:00:

Panel Discussion

What comes after graduation?
An outlook to professional work in an international academic or industry R&D context …

… and what to consider when deciding for the final thesis project

Panelists:

- Torvald Mårtensson, Saab (Linköping) and part-time associate professor at IDA
- Daniel Varro, professor at IDA and spin-off founder
- Olaf Hartig, senior associate professor at IDA and scholar at Amazon Web Services
- August Ernstsson, associate professor at IDA and recently visiting postdoc researcher at WWU Münster, Germany

Moderator:

- Christoph Kessler, LiU

Panelist Presentations (10 min. each)

Lead Questions

- Please give a short description of the organization(s) you present.
  with focus on: industry R&D (Torvald, Olaf), spin-offs (Daniel), international academic research (Daniel, Olaf, August)
- How is R&D work in the organization usually done,
  esp. in an international context? (if applicable)
  e.g., international projects, cooperation with universities or research institutes, student internships, ...?
  What is the typical time frame and size for a technical development project?
- What are typical R&D-related work activities of an engineer with a CS/CE master degree in your organization (/division/group) during the first years?
- What are typical career paths for R&D (CS/CE) engineers in your organization?
- You all have a degree from LiU and/or supervise thesis students at LiU.
  What advice can you give to final-year students
  (a) for the master thesis project (topic choice, project work), and
  (b) for later?

(Panelist statements first; Student questions will be jointly answered at the end)

Questions to panelists from the audience

LINKÖPINGS UNIVERSITET

Acknowledgements

- Most slides of the first part (thesis types) are based on a previous version courtesy of Ola Leifler, IDA, Linköping University

- Slides of some of the panel presentations will be made available in the Lisam cooperative area

Some advice for later

MOVED TO LECTURE 6

- Most career paths are *employment-based* (i.e., *dependent* employee)
  - In industry, your company through your manager(s) sets the goals (e.g. what to work on)
    - And as a self-employed person, you instead depend on your customers …
  - In academic research, "academic freedom" only relates to the choice of research area / problems, which is still constrained e.g. by external funding programmes and university priorities, and to the presentation of research results, which also follows some rules (e.g. peer-review)
  - Permanently work on your **employability**:
    Keep your only *weapon* as a dependent employee
    (= the option – or need – of changing the employer) *sharp*!
    - Keep a "red thread" in your CV
      - how will your planned next move look like in your CV when applying again a few years later?
    - Try to avoid very short employment periods (and unemployment periods)
    - Keep a good professional relation to your boss
      - you will need him/her later for a good reference
    - Try to be flexible (e.g., w.r.t. work location)
    - Collect merits (esp. in academia)
    - Never stop learning!

Source: Heiko Mell, weekly column on career advice for engineers in *VDI-nachrichten*, and books on career planning [in German]
https://www.ingenieur.de/karriere/arbeitsleben/heiko-mell