Seminar: Advanced Software Engineering

Special Focus: Cloud Computing

Dr. Philipp Leitner

University of Zurich, Switzerland

Universität Zürich[UZH]

s.e.a.l.
software evolution & architecture lab

Lecturers

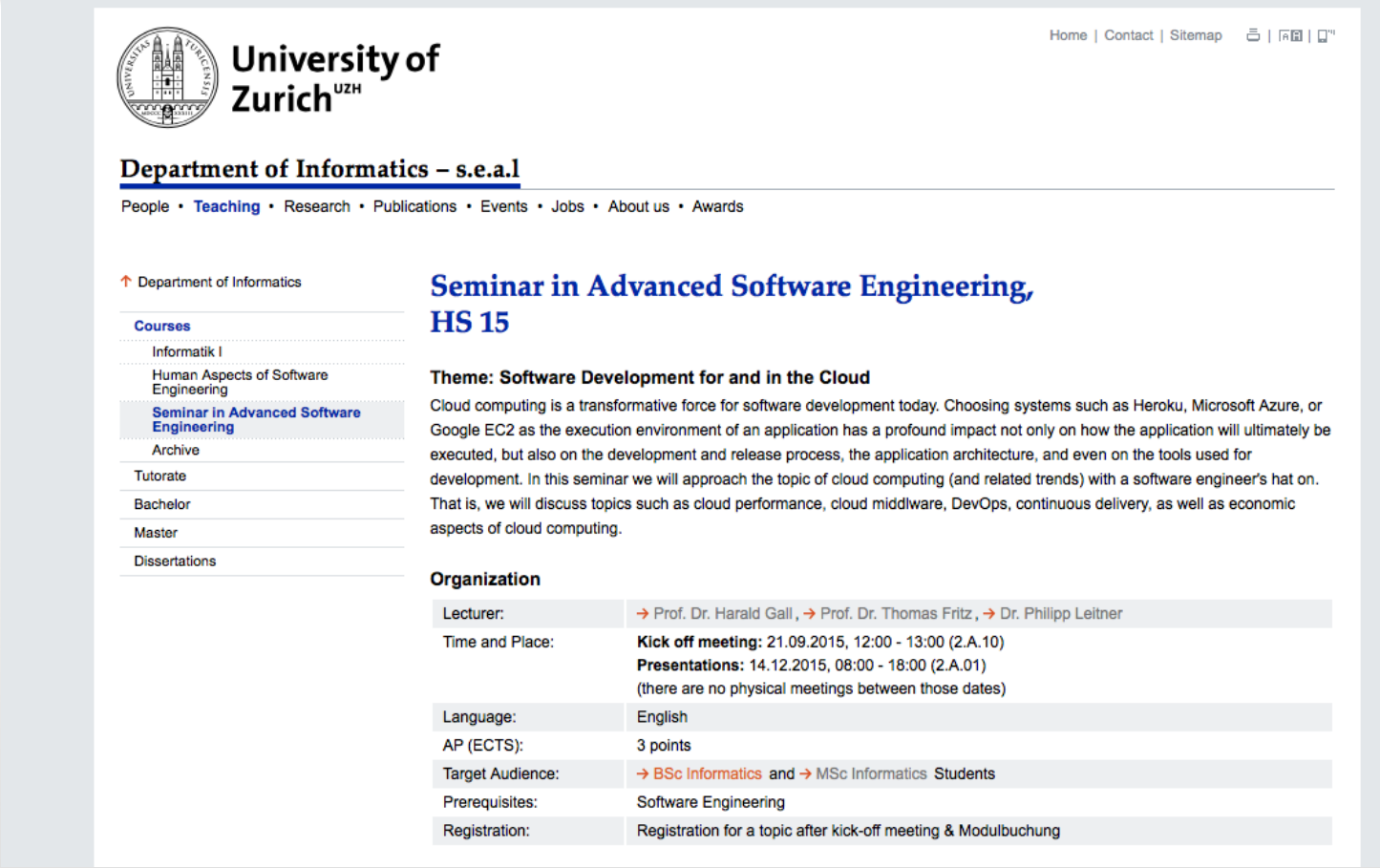
What's a seminar?

Scientific, literature-based work

Learning goals:

- *Finding* and reading good scientific literature
- Academic writing (how to summarize main points, how to cite literature, etc.)
- Giving academic presentations

All Info



University of Zurich[UZH]

Home | Contact | Sitemap

Department of Informatics – s.e.a.l

People • Teaching • Research • Publications • Events • Jobs • About us • Awards

↑ Department of Informatics

- Courses
  - Informatik I
  - Human Aspects of Software Engineering
  - Seminar in Advanced Software Engineering
  - Archive
- Tutorate
- Bachelor
- Master
- Dissertations

Seminar in Advanced Software Engineering, HS 15

Theme: Software Development for and in the Cloud

Cloud computing is a transformative force for software development today. Choosing systems such as Heroku, Microsoft Azure, or Google EC2 as the execution environment of an application has a profound impact not only on how the application will ultimately be executed, but also on the development and release process, the application architecture, and even on the tools used for development. In this seminar we will approach the topic of cloud computing (and related trends) with a software engineer's hat on. That is, we will discuss topics such as cloud performance, cloud middlware, DevOps, continuous delivery, as well as economic aspects of cloud computing.

Organization

Lecturer:	→ Prof. Dr. Harald Gall, → Prof. Dr. Thomas Fritz, → Dr. Philipp Leitner
Time and Place:	**Kick off meeting:** 21.09.2015, 12:00 - 13:00 (2.A.10) **Presentations:** 14.12.2015, 08:00 - 18:00 (2.A.01) (there are no physical meetings between those dates)
Language:	English
AP (ECTS):	3 points
Target Audience:	→ BSc Informatics and → MSc Informatics Students
Prerequisites:	Software Engineering
Registration:	Registration for a topic after kick-off meeting & Modulbuchung

http://www.ifi.uzh.ch/seal/teaching/courses/semadvse.html

General Process

1. Students select a topic (deadline this Sunday)
2. Students write a class paper (deadline November 9th)
3. Reviews (deadline November 20th)
4. Students revise paper and prepare a presentation
5. Final presentations (December 14th)
6. Final submission (December 18th)

Topic Selection

- I will present the options today
- You send me **until this Sunday** your choices of topic
  - Send me a list of 5 topics in order of preference
  - leitner@ifi.uzh.ch
  - If you don't send me a choice, you get the leftovers
- I will do some matchmaking and send you concrete topics the week afterwards

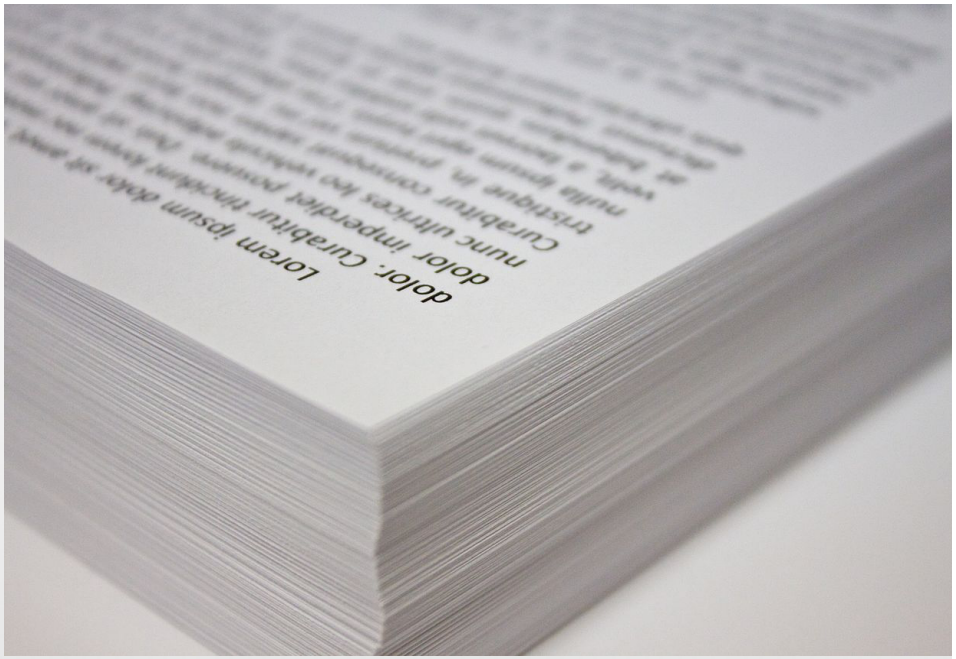
Paper

- Length: 12 - 15 pages
- Written in Latex (or, *if you really want*, in Word) using the LNCS style (link on Web page)
- English
- Needs to cite academic literature (7 - 10, typically)

Academic Literature

What's academic literature?

- Journals
- Conference proceedings
- Workshop proceedings
- (Some) books



Rule:

- Needs to be peer-reviewed (no Wikipedia / blog posts)
- Should be the original source of a claim / result

How to find academic literature?

- Google Scholar: https://scholar.google.ch
  - **Good**: gives you access to lots of literature, easy keyword search, ranks papers by citations (~ importance)
  - **Bad**: hard to find something useful if you don't know yet what you are looking for, much crap can be found, tends to prefer older sources
- By looking through relevant venues (conferences, journals)
  - **Good**: does not overwhelm you so much, more quality control, newer literature, more "explorative" (easier to find something you didn't think of so far)
  - **Bad**: more cumbersome, you need to know what the relevant venues are
  - http://dblp.uni-trier.de (indexes all important CS literature)
- "Reference crawling"
  - Find good start points, follow forward and backward references
  - Tip: Google Scholar can show you forward references ("What papers cite this?")

Accessing academic literature

- Publications can be either **open access** or behind a paywall
- Publications behind a paywall can usually be accessed freely from the UZH or ETH network
  - Either physically sit at UZH or use the VPN services
- For some important venues, UZH does not have a subscription.
  - Try googling the paper. Many “preprint” versions are available freely on the Web.
  - Otherwise: check ETH, check ResearchGate (http://researchgate.net/), or mail the authors

Writing your paper

For questions of style:

- Reflect how the **good** papers you are reading are written:
  - Avoid unnecessary and unwarranted superlatives
  - Avoid claims that are not supported by data
  - Avoid colloquial language
  - Prefer passive over active voice

Writing your paper

Rules of academic honesty apply!

- **No plagiarism!**
  - You can't copy *anything*. All text / figures need to be your own.
  - Exception: direct quotations (use *very* sparingly, visually distinct from text, immediately followed by citation)
- **No misrepresentation!**
  - Never claim something that is not actually supported by your references.
- **Make clear who is the source of what!**
  - Don't just cite a source randomly on a page and assume that the reader infers that the rest of this part of the text is based on this source.

Submission

We use Easychair to submit papers and reviews

https://easychair.org/conferences/?conf=seminarase15

(you'll need an account, it's free)

Reviews

Every paper will be reviewed by a lecturer and 2 -3 students.

Reviews should comment on:

Technical quality
Logical structure
Presentation
Style
References

Each category should be graded on an A to D scale:

A: An excellent work.
B: A good work with just a couple of small weaknesses.
C: An average work with clear weaknesses.
D: Insufficient work with many substantial weaknesses.

Reviews are to be submitted via Easychair as well.

Presentations

All presentations on a single day

Attendance is mandatory!

Per student:

- **20 minutes presentation**
- **10 minutes discussion**

Usage of Powerpoint / Keynote / Beamer is suggested.

(Prezi not so much)

Grading

Final grade will be given out after all reports are submitted.

Based on (in order of importance):

- Quality of class paper
- Quality of presentation
- Quality of reviews (the ones you write, *not* the ones you receive)
- Participation during the presentations (asking questions etc.)

Not following rules of academic honesty == auto-fail

Topics

Topic 1: Cloud Adoption and Usage

How and why are developers and companies choosing to use (or not use) cloud applications for provisioning applications? What does this decision impact, and how?

Topic 2: Cloud Benchmarking

How can developers (and researchers) evaluate the performance of IaaS cloud services? What have been the main observations of existing cloud benchmarking studies?

Topic 3: Selecting the Right Cloud Service(s)

How can developers select the best cloud service, or combination of cloud services, for their applications? What are the primary factors that need to be taken into account?

Topic 4: Pricing

How do cloud providers determine the price of instances or other types of cloud resources? Is the price always fixed? What are the main factors that cloud users are priced on?

Topic 5: Security and Privacy

What are the main challenges related to security and privacy in the context of cloud computing? What specific security and privacy challenges does cloud computing raise, and what are known successful attacks?

Topics

Topic 6: Modelling Cloud Applications
What standards and other proposals are there for modelling the structure and cloud deployment of applications? What do these proposals have in common, and how do they differ?

Topic 7: Client-Side Middleware
What client-side libraries, frameworks, and middleware systems are there for supporting the development of cloud applications? What functionality do these systems typically provide, and how do they differ?

Topic 8: Platforms
What server-side frameworks, and middleware systems are there for deploying cloud applications? What functionality do these systems typically provide, and how do they differ?

Topic 9: Developer Performance Awareness
How can cloud developers monitor the performance of their applications? What are important metrics to keep track of?

Topic 10: Releasing Changes Fast(er)
How do faster releases, as often associated to cloud computing, impact the quality of software?


why
how
when
which
what
where
who