



Seminar Techniques

Tools for the Application of Formal Methods in Software
Development

Dr. Michael Petter

Summer Semester 2015

Title: Petter: Seminar Introduction (24.04.2015)

Date: Fri Apr 24 13:06:30 CEST 2015

Duration: 46:47 min

Pages: 16

Organization

Deadlines

- ▶ Start: with researching topic background
- ▶ May 11th: Outline of the seminar paper/seminar talk
- ▶ May 18th: Q&A session for topic research
- ▶ June 5th: Extended abstract for seminar paper & Talk slides
- ▶ July 10th: Final Paper

Proceeding

- ▶ 2 days of talk – attendance compulsory
- ▶ Talks in sessions of 30 minutes
 - ▶ Extended abstracts to be distributed
 - ▶ 25 minutes of talk (sharp!)
 - ▶ 5 minutes of questions
- ▶ Full paper at the end of the semester

What is the common idea of this seminar?

Don'ts

- ▶ This is not a marketing event
- ▶ You are no historian
- ▶ Do not despair in the face of algorithmic challenges

Do's

- ▶ This is a **scientific** talk
- ▶ The art is to explain the details of a complex solution to the audience
- ▶ You can expect a generic CS-educated audience

Talk preparation

- ▶ A seminar talk is **not a lecture!**
- ▶ *Spoken words* dominate
- ▶ Do **not** begin to plan your talk by drawing slides!
- ▶ Instead:
 - ▶ Concentrate on **core ideas!**
 - ▶ Plan your talk strategically to culminate in the core ideas **keep to a central theme**
 - ▶ Try not do get distracted
- ▶ Most good talks resemble a story
- ▶ Make sure that you head for the core ideas during the talk
- ▶ Check whether you succeed in transmitting these messages to a **test audience**

How do I prepare a good talk?

Slides

Purpose: Support the presenter during his/her talk

- ▶ **Use slides for:**
 - ▶ saving drawing time for figures/charts/source code
 - ▶ replacing complex panel paintings
 - ▶ provide an overview
 - ▶ new slides for new topics
 - ▶ simplifying complex diagrams/figures
- ▶ **No-Goes**
 - ▶ **bulletlisting in great profusion**
 - ▶ **continuous floating text**
 - ▶ overloaded diagrams
 - ▶ reproducing every detail you say
 - ▶ **more than 10 lines**

Immediately before

Technical Aspects

- ▶ Slides in PDF-Format on a USB-key as backup
- ▶ Testing the own notebook at home or in university at the VGA port in advance
- ▶ No widescreen nor high-resolution modes; use: XGA (1024x768)

Talk

- ▶ calculate min. 2-3 Minutes per slide
- ▶ **Sample talk** within a sharp time frame
- ▶ **Sample talk** with a test audience obligatory
- ▶ optional slides in the appendix

How do I create a decent paper?

During talk

- ▶ talk **freely**, not reading slide contents
- ▶ not display a set of **unrelated** slides
- ▶ **Central theme** needs to shine through; communicate
 - ▶ Cause/Effect
 - ▶ Problem/Solution
 - ▶ Alternatives
- ▶ Consider previous knowledge
- ▶ Involve the audience
 - ▶ Body language
 - ▶ Language
 - ▶ Eye contact
 - ▶ Dialogue instead of monologue

Working with sources

Non-scientific

- ▶ Internetarticles/Blogs (e.g. heise Newsticker)
- ▶ Computermagazines (e.g. c't)
- ▶ Lexica (e.g. Wikipedia)

Nice for motivation/overview; → *Reliability?*

Scientific

Sogenannte "Paper" aus

- ▶ Technical reports, "White Papers"
- ▶ Scientific magazines
- ▶ Conference proceedings

Research via DBLP (<http://www.dblp.org>), Google Scholar and TUM-Library

Important

- ▶ Structure: Demonstrate the problem, Solve the Problem, Evaluate solution
- ▶ Literature register
- ▶ Own statements vs. citations
- ▶ Keep attention to the quality of figures!

Form: Template for Latex/Word-Documnt in PDF

Extended abstract

1. ≤ 3 pages
2. gives a rough overview
3. motvation and ≤ 3 central concepts

Final Paper

1. ≤ 10 pages
2. feedback from the audience
3. detailed explanations of the central concepts
4. critical evaluation

Important

- ▶ Structure: Demonstrate the problem, Solve the Problem, Evaluate solution
- ▶ Literature register
- ▶ Own statements vs. citations
- ▶ Keep attention to the quality of figures!

Form: Template for Latex/Word-Documnt in PDF

Extended abstract

1. ≤ 3 pages
2. gives a rough overview
3. motvation and ≤ 3 central concepts

Final Paper

1. ≤ 10 pages
2. feedback from the audience
3. detailed explanations of the central concepts
4. critical evaluation

What is my topic?

What is my topic?

Assigned Topics

1. Whitebox fuzzing with Java Path Finder
2. Verifying the Absence of Floating Point Errors with Astree Runtime Analyzer
3. Data Race Detection with Locksmith
4. Automatic Optimal Fence Placement in Relaxed Memory Models with Fender
5. Verifying Object Oriented Programming Languages with MS Dafny and Boogie/Z3
6. Programming with Contracts and Verifying them with MS Spec#
7. Analyzing the the Reachability of Error States with CPAchecker
8. Analyzing Dynamic Data Structures with Predator
9. Analyzing Interactions of Real Time Systems with UPAAL
10. Verifying a C Compiler via Theorem Prover with Coq

Replacement Topics

1. Static and Dynamic Java Bytecode Analysis with Chord
2. Dynamic Analysis of Java Bytecode with DiSL/ShadowVM
3. Static Analysis to Create Parallelization Candidates with Sambamba
4. Symbolic Execution with KLEE/LLVM
5. Analyzing Security Properties of Probabilistic Models (esp. communication protocols) with PRISM