Christian Berger

christian.berger@cse.gu.se

Towards Thinking Cars

General topics of interest:

- Autonomous driving for various platforms (1:10, cars, trucks)
- Continuous deployment and experimentation for automotive systems
- Simulations for autonomous driving solutions

Carolo-Cup

Dr. Thorsten Berger

thorsten.berger@cse.gu.se

- Research on Highly Configurable Systems (HCS)
  - Software Product Lines
  - Software Ecosystems

- In the Domains of
  - Systems Software
  - Automotive, Avionics, and Embedded Systems
  - Mobile Apps

- Developing Methods and Tools for
  - Adopting and Evolving HCS
  - Modeling and Configuring HCS

Jan Bosch

jan.bosch@chalmers.se

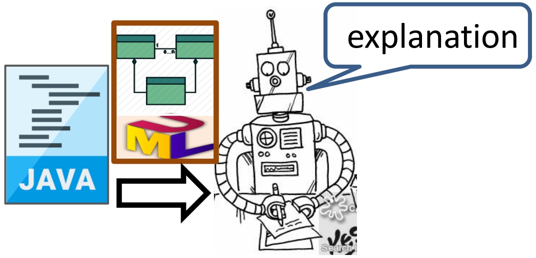
General interests:
- Software architecture and platforms
- Evidence-driven development
- Software ecosystems
- Innovation and startups

Michel Chaudron

chaudron@chalmers.se

General Interests:

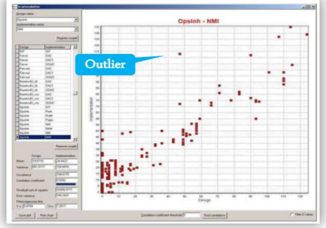
Software Design and Modeling
Automated Program Understanding



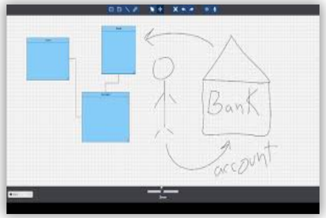
Thesis Project suggestions:

- Automated Program Explanation = AI for program understanding
  To design and build software that can explain (other) software based on the automated analysis of software project artefacts such as: UML design + source code (including comments) + commit message + tests + …
- Analysing the correspondence between design and code over time
- Analysing the impact of architecture design on source code modularity
- Analysing the relation between source code commenting patterns and importance of classes in software design (BSc)
- Automated Lay-outing of Class Diagrams
  - Based on: class-role, importance, user preferences, machine learning from examples
- Software Design Environment on a Whiteboard
  - Support distributed awareness in remote collaboration

Ho Quang Truong



Rodi Jolak



Ivica Crnkovic

crnkovic@chalmers.se

- component-based software engineering
- software architecture
- software configuration management
- software development environments and tools
- software engineering in general

Regina Hebig

hebig@chalmers.se

- Software Evolution
  - Software Comprehension
  - Software Clones and Refactoring
  - Software Maintenance & Changeability
  - System Migration and Updating
- Model-Driven Engineering (MDE) & Low-Code Development
- Hybrid Software Processes
- Digital certification in additive manufacturing (Co-operation with SWEREA)

Jennifer Horkoff

jenho@chalmers.se

@jenhork

- **Requirements Engineering & Early Requirements Modeling**
  - Goal Modeling and Creativity (http://creativeleaf.city.ac.uk/)
  - Business Intelligence Modeling (http://www.cs.utoronto.ca/~jm/bim/)
- **Requirements Modeling and Game Development (with S. Björk)**
- **Strategic API Value and Measurements**
  - (Software Center Project #26)
  - With Axis, Bosch, Ericsson, Grundfos, & Tetra Pak
- **Large-Scale Agile Requirements Engineering**
  - (Software Center Project #27, with Eric Knauss)
  - With Bosch, Ericsson, Grundfos, Siemens, Tetra Pak, Volvo Cars, & Volvo Trucks

Full academic page: www.cs.utoronto.ca/~jenhork

Francisco Gomes

gomesf@chalmers.se

Research interests:

- Software **Testing Techniques**
  - Test case selection, minimization, prioritization;
- (Meta-)**Empirical** Software Engineering
  - Reproducibility, replication, re-analysis;

Ideas for thesis proposal:

- Platform for **automated experiments** with software testing techniques;
- A **formal experiment** to investigate disparate software testing techniques;
- Integration between test tools and **automated testing techniques**;
- **Meta-heuristics** and synthesis test artefacts through meta-heuristics;
- **Reproducible research!**

What will we explore **together**?

- Statistics, quantitative research, test processes, development of tools.

Eric Knauss

eric.knauss@cse.gu.se

Main topics:

- Requirements Engineering
- Agile Methods

Especially when applied to:

- Global Software Development
- Continuous Integration and Deployment
- Software Ecosystems

Philipp Leitner

philipp.leitner@chalmers.se

@xLeitix

- Areas and keywords:
  - **Cloud computing** (AWS, Google, …)
  - **Services computing** (Microservices, …)
  - **Performance monitoring / testing** (load testing, JMH, …)
  - **Deployment and middleware** (Docker, Kubernetes, …)
  - **Continuous experimentation** (A/B testing, canaries, …)
  - … many other things that relate to SE for Web systems

- Types of theses:
  - Systems thesis → build and evaluate a cool prototype
  - Empirical thesis → form an interesting hypothesis, test it on data

- More info: http://philippleitner.net/theses/

Grischa Liebel grischa@chalmers.se

- Model-Driven Engineering (MDE)
- Requirements Engineering
  - Integration of Requirements Engineering into MDE
- Requirements Communication & Knowledge Management
- Tool Interoperability

- Check my publications to get an overview over my topics and possible connection points!

Agneta Nilsson

- Change Management / Software Process Improvement
  - Organizing and managing ways of working – transformations
  - From Traditional to Agile and Continuous Deployment
- User Experience
  - Integrating UX practices into SE practices

Patrizio Pelliccione

Verification Modeling

Modeling and Verification of Software Architectures

Interoperability

Interoperability among tools and languages

Model elicitation

Automatic construction of models

Autonomous quadrotors

Riccardo Scandariato

riccardo.scandariato@cse.gu.se

- **Security**
- **Privacy**

Miroslaw Staron

Measuring organizational performance

- Industrial problem formulation
  - How to use existing formal and informal metrics at the partner companies and the role these metrics play in decision making (as well the quality of these decisions)?
- Approach in the project
  - Mine data from source code, defect repositories, effort reporting systems
  - Compile the data into ISO 15939 indicators & measurement systems
- Milestones/results so far
  - Objective release-readiness indicators
  - Source code risk assessment using heatmaps
  - Product stability assessments before release

Dr. Jan-Philipp Steghöfer
jan-philipp.steghofer@cse.gu.se

- Traceability Management
  How can you figure out what you need to change in your design/code/tests/requirements when something changes in your project?

- Variability
  How can you handle the different variations of a hardware and software in a large embedded systems product?

- Agile Processes for Self-Adaptive Systems
  How do you build a system in an agile fashion that self-adapts to the environment?

Richard Torkar

Richard.Torkar@cse.gu.se

- Software testing
- Software quality
- Applied statistics
  - Bayesian statistic
  - Machine learning