UNIVERSITY OF AMSTERDAM

EU Erasmus+
Strategic Partnership for Higher Education

Focusing Education on Composability, Comprehensibility and Correctness of Working Software (3COWS)

Programme

First Teacher Training

Amsterdam, December 11-15, 2017

Clemens Grelck

Co-funded by the
Erasmus+ Programme
of the European Union

Monday, December 11, 2017

	What	Who	Where
12:00 – 12:30	Welcome	Ana + Clemens	UB
12:30 – 14:00	Lunch		UB
14:00 – 15:00	User-centric Scheduling on Clouds	Ana	UB
	and Clusters		UB
15:00 – 16:00	CodeCompass	Zoltan + Tibor	UB
16:00 – 16:30	Coffee break		UB
16:30 – 18:00	CodeCompass	Zoltan + Tibor	UB

Tuesday, December 12, 2017

	What	Who	Where
09:00 – 09:15	Welcome at Vrije Universiteit	Ana + Clemens	VU
09:30 – 11:30	Visit Green IT Lab	Patricia + Ivano	GITlab
11:30 – 12:30	Coffee, Discussion and Brainstorming	Patricia + Ivano	GITlab
12:30 – 14:00	Lunch and campus visit	Patricia + Ivano	VU
14:00 – 14:30	Travel to city centre	Ana + Clemens	
14:30 – 16:00	Canal tour	Ana + Clemens	
16:00 – 17:30	Energy Efficiency across Programming	João	OMHP
	Languages		OMHP
17:30 – 19:00	Standard ML Structures and Functors	Štefan	OMHP
	in CPN Models		OMHP

Wednesday, December 13, 2017

	What	Who	Where
09:30 – 10:30	Energy Efficiency in Regression Testing	Csaba	UB
10:30 – 11:00	Coffee break		UB
11:00 – 12:00	Nijmegen Devices Lab	Rinus + Pieter	UB
12:00 – 13:00	Lunch		UB
13:00 – 15:00	Gamification in Teaching	Alexandru	UB
15:00 – 15:30	Coffee break		UB
15:30 – 17:00	Status Update Winter School	Csaba, all	UB
17:00 – 18:00	Intellectual Output Teacher Training	Clemens, all	UB
18:30 – 21:00	Joint Dinner		KT

Thursday, December 14, 2017

	What	Who	Where
09:30 – 10:00	Teaching innovation at UvA	Natasa + Clemens	SCP-B
10:00 – 10:30	Digital testing: experience report	Sebastian	SCP-B
10:30 – 11:00	Iupiter Notebooks in practice	Maarten	SCP-B
11:00 – 11:30	Coffee break		SCP-B
11:30 – 12:45	Visit Knowledge Clips Recording Studio	Natasa	SCP-B
13:00 – 14:00	Lunch		
14:15 – 15:15	Blended Learning	Natasa	SCP-B
15:15 – 15:45	Coffee		SCP-B
15:45 – 16:30	Blended Learning at Winter School?	All	SCP-B

Friday, December 15, 2017

	What	Who	Where
09:30 – 09:45	Introduction: How meta-programming helps to achieve Composability, Comprehensibility and Correctness of Working Software	Jurgen, Paul, Tijs	CWI
10:15 – 11:00	Can software survive the Information Society?	Paul	CWI
11:00 – 11:15	Coffee		
11:15 – 12:00	What if your code were data?	Jurgen	CWI
12:00 – 13:00	Lunch		CWI
13:00 – 13:45	Domain Specific Languages in Practice	Tijs	CWI
13:45 – 14:30	Reflection versus Static Analysis	Jurgen	CWI
14:30 – 15:30	Discussion: Does meta-programming deliver on its promises for forward and reverse engineering?	All	CWI
15:30 – 16:30	Farewell Drinks		Polder

Locations

Name	Information
UB	Amsterdam University Library (Dutch: Universiteitsbibliotheek) Address: Singel 425 Tram: Koningsplein, lines 1, 2, 5 Room: Potgieterzaal Instructions: pass through barriers (to be ignored), turn left, turn left again
VU	VU University Amsterdam (Dutch: Vrije Universiteit) Address: De Boelelaan 1105 Metro: De Boelelaan / VU, line 51 Room: Coffee corner in main building (Dutch: hoofdgebouw)
GITlab	VU University Amsterdam (Dutch: Vrije Universiteit) Address: De Boelelaan 1105 Metro: VU / De Boelelaan, line 51 Room: HG-13A33
OMHP	University of Amsterdam Oudemanhuispoort Complex Address: Oudemanhuispoort 4–6 Room C0.23
KT	Restaurant Kantjil en de Tijger Address: Spuistraat 291–293
SCP-B	University of Amsterdam Science Park Campus Address: Science Park 904 Train: Amsterdam Science Park, Sprinters to Amersfoort, Zwolle Bus: Science Park, lines 40, 240 from Muiderpoort and Amstel stations Room: B1.25 (beta lounge)
CWI	Centrum Wiskunde en Informatica Address: Science Park 123 Train: Amsterdam Science Park, Sprinters to Amersfoort, Zwolle Bus: Science Park, lines 40, 240 from Muiderpoort and Amstel stations Room: L120

Consortium Participants

Name	Affiliation	Country
Ana Oprescu	University of Amsterdam	Netherlands
Clemens Grelck	University of Amsterdam	Netherlands
Csaba Szabó	Technical University of Košice	Slovakia
Elena Somova	University of Plovdiv	Bulgaria
João Saraiva	University of Minho	Portugal
João Paulo Fernandes	University of Coimbra	Portugal
Judit Robu	Babes-Bolyai University of Cluj-Napoca	Romania
Pieter Koopman	Radboud University Nijmegen	Netherlands
Rinus Plasmeijer	Radboud University Nijmegen	Netherlands
Štefan Korečko	Technical University of Košice	Slovakia
Tibor Brunner	Eötvös Loránd University Budapest	Hungary
Tihana Galinac Grbac	University of Rijeka	Croatia
Viktória Zsók	Eötvös Loránd University Budapest	Hungary
Zoltán Porkoláb	Eötvös Loránd University Budapest	Hungary

Amsterdam Area Contributors

Name	Affiliation	Country
Alexandru Iosup	VU University Amsterdam	Netherlands
Huub Rutjes	University of Amsterdam	Netherlands
Ivano Malavolta	VU University Amsterdam	Netherlands
Jurgen Vinju	Centrum Wiskunde en Informatica	Netherlands
Maarten Marx	University of Amsterdam	Netherlands
Natasa Brouwer	University of Amsterdam	Netherlands
Patricia Lago	VU University Amsterdam	Netherlands
Paul Klint	Centrum Wiskunde en Informatica	Netherlands
Sebastian Altmeyer	University of Amsterdam	Netherlands
Tijs van der Storm	Centrum Wiskunde en Informatica	Netherlands

Presentations

Ana Oprescu:
User-centric Scheduling on Clouds and Clusters

Commercial cloud providers offer compute resources with defined quality of service (CPU type and clock speed, size of main memory etc). These compute resources can be rented through various pricing models. While these commercial offerings come with clear machine specification, users still need guidance when choosing among them, simply because most of the time these specifications say little about the performance of a given application. Our research project, BaTS, addressed such user-centric issues as resource selection and application scheduling on clouds. Based on our findings, we designed educational activities underpinning a master level course.

Zoltán Porkoláb, Tibor Brunner:
CodeCompass

Bugfixing or new feature development requires a confident understanding of all details and consequences of the planned changes. For long existing large telecom systems, where the code base have been developed and maintained for decades by fluctuating teams, original intentions are lost, the documentation is untrustworthy or missing, the only reliable information is the code itself. Code comprehension of

such large software systems is an essential, but usually very challenging task. As the method of comprehension is fundamentally different from writing new code, development tools are not performing well.
During the years, different programs have been developed with various complexity and feature set for code comprehension but none of them fulfilled our special requirements. CodeCompass is an open source LLVM/Clang based tool developed by Ericsson and the Eötvös Loránd University, Budapest, to help understanding large legacy software systems. Based on the LLVM/Clang compiler infrastructure, CodeCompass gives exact information on complex C/C++ language elements like overloading, inheritance, the (read or write) usage of variables, possible calls on function pointers and the virtual functions - features that various existing tools support only partially. The wide range of interactive visualizations extends further than the usual class and function call diagrams; architectural, component and interface diagrams are a few of the implemented graphs. To make comprehension more extensive, CodeCompass is not restricted to the source code. It also utilizes build information to explore the system architecture as well as version control information when available: git commit history and blame view are also visualized. Clang based static analysis results are also integrated to CodeCompass. Although the tool focuses mainly on C and C++, it also supports Java and Python languages. Having a web-based, pluginable, extensible architecture, the CodeCompass framework can be an open platform to further code comprehension, static analysis and software metrics efforts.
https://github.com/ericsson/codeCompass

João Saraiva:
Energy Efficiency across Programming Languages: How Do Energy, Time, and Memory Relate?

In this talk we present a study of the runtime, memory usage and energy consumption of twenty seven well-known software languages. We monitor the performance of such languages using ten different programming problems, expressed in each of the languages. Our results show interesting findings, such as, slower/faster languages consuming less/more energy, and how memory usage influences energy consumption. We show how to use our results to provide software engineers support to decide which language to use when energy efficiency is a concern.

Štefan Korečko:
Standard ML Structures and Functors in CPN Models

Coloured Petri Nets (CPN) represent an interesting example of functional programming techniques and methodologies utilization in a specific application area. CPN are a formal language that models discrete-event systems as bipartite, directed graphs. It provides an easy to understand notation for distributed states, non-determinism, concurrency and parallelism. In addition, there is a mature software tool, called CPN Tools, for specification, simulation and verification of CPN models. CPN, as implemented in CPN Tools, use the functional programming language Standard ML (SML) to describe data manipulation and state transition conditions. Consequently, the full expressive power of functional programming technology is at the modelers disposal.
In this lecture the participants will learn how to use CPN and SML for the modeling and simulation of stochastic timed event-driven models. An original model, based on the ideas and suggestions of the participants, will be constructed. The model will use a modular specification in SML, using the structure and functor constructs.

Csaba Szabó:
Introducing Energy Efficiency Measurements into (Regression) Testing

We present how we aim at creating a habit/practice of energy consumption measuring as common regression testing activity. The students are evolving their software, and besides the common strategies of following code metrics to minimize refactoring actions, re-testing and regression testing they are

required to measure energy consumption of their test cases and write verdicts about changes in energy consumption as well.

Rinus Plasmeijer, Pieter Koopman: Nijmegen Devices Lab

In this presentation we will give a brief overview of Task Oriented Programming (TOP) the Internet of Things (IoT) and their connection. The main purpose of this session is to decide about the goals and content of the teacher training in Nijmegen.

Public Transport in Amsterdam

You may need a fair deal of public transport to commute between your hotel and the various locations of the teacher training. Public transport in Amsterdam is extremely well organised, reasonably priced for locals and Dutch residents and very expensive for tourists.
Individual tickets are highly overpriced and no longer sold for cash. You need a debit card that works in the Netherlands (and doesn't incur prohibitive fees back home).
While there are 24/48/72-hour tickets, mainly sold to tourists in hotels, these only pay off after many individual trips. And, after all, the center of Amsterdam is not too big to walk to most places.
In the Netherlands we have a nation-wide electronic ticketing system named *OV-ChipCard*. The OV stands for public transport (Dutch: openbar verwoer). The advantage of this system is that it works everywhere across the entire country. So, you can reuse your card when you attend the second teacher training in Nijmegen. The best option is to buy one first thing upon arrival, either at the airport or the Centraal station at the very latest. The cost is EUR 7.50 for the card itself plus whatever amount you want to load it with.
These cards are available from ticket vending machines, but also from counters at Schiphol airport or the service center of the Amsterdam public transport provider GVB at Centraal station. When leaving the station building, watch out for a beautiful wooden building in front of you. Talking to a human being might be easier than fighting with a vending machine. English will be spoken wherever you go. You can later use the machines of GVB or the railway operator NS to top up your card.
Whenever you enter a bus/tram/metro, you must check-in at a device. Important: whenever you leave a bus/tram/metro you must check-out at another device. If you forget to check-in, someone will shout at you. If you forget to check-out, you pay the maximum possible amount and so quickly drain your card. Beware: theoretically you can get your money back under certain circumstances, but that in any case requires a Dutch bank account. You understand the systematics: we always want the best from our guests.
Another important message: always board a bus through the front door, always leave it through one of the other doors. Trams have dedicates entrance and exit doors. There are usually (depending on the model) two entrance doors: at the front and about one third length from the back.
In principle the OV-ChipCard also works for trains anywhere in the country, but you must first activate your card for rail transport, and, more importantly, you need a fairly substantial minimum balance to board a train. So, I would rather discourage that use and only use the card for metro, trams and buses. If you only use the trains from and to the aiport, it is better to buy individual tickets, even though you must pay extra for the plastic that stores your ticket because when you leave the country with a substantial balance on your card that usually constitutes a much bigger loss.