Master's Specialisation in Software Science

Frits Vaandrager

Radboud University

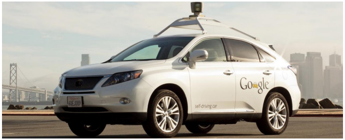
November 2018

1 Master's in Computing Science

- degree: Master of Science (MSc)
- duration: 2 years (120 EC), full-time
- specialisations:
  - *software science*
  - cyber security
  - data science
  - MFoCS
- software science: for whom?
  - you are passionate about programming
  - you are interested in scientific questions surrounding software
  - you are interested in building systems
  - you are creative
- career prospects: simply *excellent*

2 The Semantic Gap

- Computers are everywhere . . .



- . . . and they seem to be able to do anything.

2 The Semantic Gap

- However, on the most basic level a computer is very primitive.
- It consists of a large collection of switches, bits, which can take one of two values: 0 or 1.
- It only supports a handful of simple operations:
  - set bit to 0 or 1;
  - test bit (**if** x==0 **then** ... **else** ...).
- What is it that transforms these trivial operations on bits into the incredible feats we see computers perform?

2 The Semantic Gap

- However, on the most basic level a computer is very primitive.
- It consists of a large collection of switches, bits, which can take one of two values: 0 or 1.
- It only supports a handful of simple operations:
  - set bit to 0 or 1;
  - test bit (**if** x==0 **then** ... **else** ...).
- What is it that transforms these trivial operations on bits into the incredible feats we see computers perform?

Software!

2 Software Science

Challenges:

- ensure correctness
- ensure security
- ensure robustness
- tame complexity
- embrace legacy
- short time to market

Facets:

- science
- technology
- creativity
- beauty

2 Software Science

Challenges:

- ensure correctness
- ensure security
- ensure robustness
- tame complexity
- embrace legacy
- short time to market

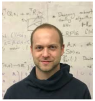
Facets:

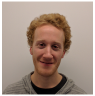
- science
- technology
- creativity
- beauty

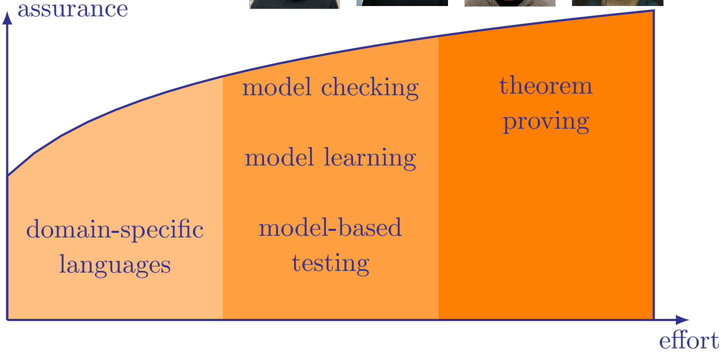
Our mission:

Develop theory, languages, and tools that simplify the construction of reliable software systems.

3 Research: High-assurance Software







3 Research: Software Technology

- functional programming

  keep the fun in programming

- generic programming

  how to let the computer do the work

- domain-specific languages

  speaking the language of domain experts

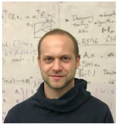
- software quality

  how do we measure it?

- *a research highlight:*

  iTasks — *task-oriented programming*

3 Research: Model Checking and Learning



- model checking

  is the bad state ever reached?

- model learning

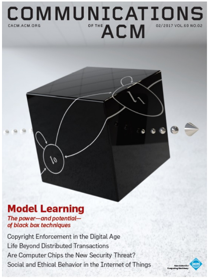
  how to reverse engineer models from code?

- model-based testing

  does a system meet its specification?

- *a research highlight:*

  cover article CACM



COMMUNICATIONS OF THE ACM

CACM.ACM.ORG

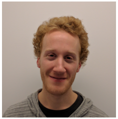
02/2017 VOL.60 NO.02

Model Learning

The power—and potential—
of black box techniques

Copyright Enforcement in the Digital Age
Life Beyond Distributed Transactions
Are Computer Chips the New Security Threat?
Social and Ethical Behavior in the Internet of Things

3 Research: Foundations



- interactive theorem proving

  proofs with the computer as an assistant
- automated theorem proving

  how to let the computer do most of the work
- type theory

  a foundation for computer science and mathematics
- *a research highlight:*

  formal semantics of C

4 Programme Outline

semester 1	track basic	track basic	track basic	track elective	free elective
semester 2	research seminar	track elective	track elective	track elective	master elective
semester 3	research internship		CS and society	master elective	master elective
semester 4	Master's project and thesis				

4 Courses

Basic courses

- Advanced Programming
- Testing Techniques
- Design of Embedded Systems

Elective Courses

- *Software Engineering:*
  - Compiler Construction
  - Software Development Entrepreneurship
  - Software Security
- *Computer-Aided Analysis:*
  - Model Checking
  - Software Analysis
  - Automated Reasoning
  - Proof Assistants
- *Theory of Computation:*
  - Type Theory & Coq
  - Semantics & Domain Theory
  - Complexity Theory
  - Computability Theory

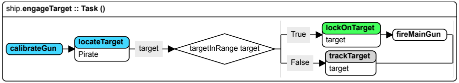
4 Course: Advanced Programming

- generic programming

  how to let the computer do the work
- design and implementation of domain-specific languages

  speaking the language of domain experts
- *application:* a DSL for tasks

  for the new Dutch navy vessel



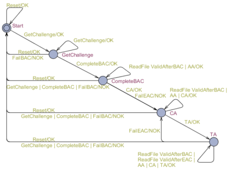
4 Course: Testing Techniques

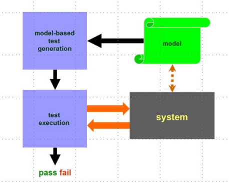
- state-of-practice testing techniques

  problems and challenges of software testing in industry
- model-based testing

  automatic generation and execution of test cases
- model learning, or test-based modelling

  reverse engineering of models for regression testing, legacy software, and security analysis


Reset/OK
Start
GetChallenge/OK
GetChallenge/OK
Reset/OK
FailBAC/NOK
GetChallenge
CompleteBAC/OK
ReadFile ValidAfterBAC | AA/OK
Reset/OK
CompleteBAC
GetChallenge | CompleteBAC | FailBAC/NOK
CA/OK
FailEAC/NOK
ReadFile ValidAfterBAC | AA | CA/OK
Reset/OK
GetChallenge | CompleteBAC | FailBAC/NOK
CA
TA/OK
FailEAC/NOK
TA
Reset/OK
GetChallenge | CompleteBAC | FailBAC/NOK
ReadFile ValidAfterBAC | ReadFile ValidAfterEAC | AA | CA | TA/OK

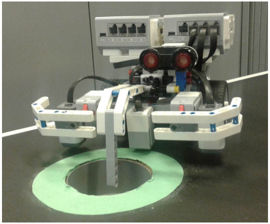


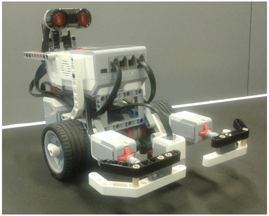
4 Course: Design of Embedded Systems

- scheduling and real-time operating systems

  exercises with Xenomai, control LED clock
- model-based development of embedded systems

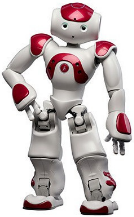
  use of domain-specific languages to develop missions for Lego EV3 Rovers



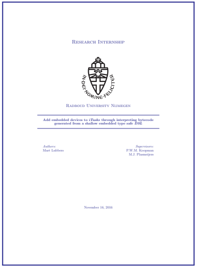


5 Research Internships

- join one of our research groups
- join our *New Devices Lab*



5 Add embedded devices to iTasks



Research Internship

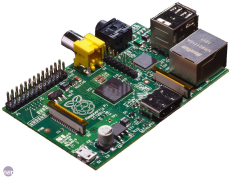
Radboud University Nijmegen

Add embedded devices to *iTasks* through interpreting bytecode generated from a shallow embedded type safe *DSL*

Authors:
Mart Lubbers

Supervisors:
P.W.M. Koopman
M.J. Plasmeijers

November 16, 2016

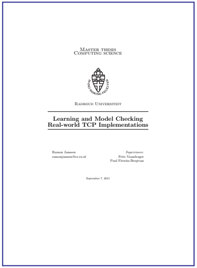


- embed *IoT* in *iTasks*
- work with embedded devices in the *New Devices Lab*

6 Master's Thesis Projects

- within our institute
- within industry & society: ASML, Axini, Altran, TomTom, Thales, Sogeti, Océ, NXP, Philips, Belastingdienst, Navy, ...
- abroad: Oxford, Uppsala, London, ...

6 Learning TCP Implementations



Master thesis
Computing science

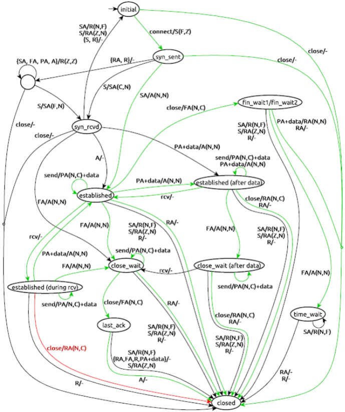
Radboud Universiteit

**Learning and Model Checking
Real-world TCP Implementations**

Ramon Janssen
ramonjanssen@cs.ru.nl

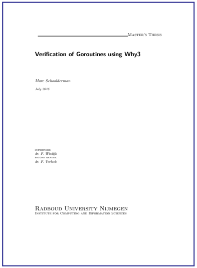
Supervisors:
Frits Vaandrager
Paul Fiterău-Broştean

September 7, 2015



- model learning to construct state machine model of TCP protocol
- presented at CAV 2016
- Windows 8 TCP client: non-conforming (buggy)

6 Verification of Goroutines using Why3



Master's Thesis

Verification of Goroutines using Why3

Marc Schoolderman

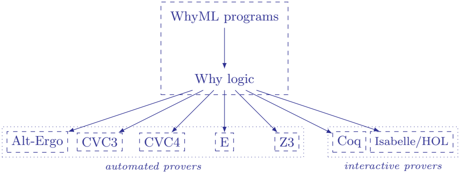
July 2016

supervisor:
dr. F. Wiedijk
second reader:
dr. F. Verbeek

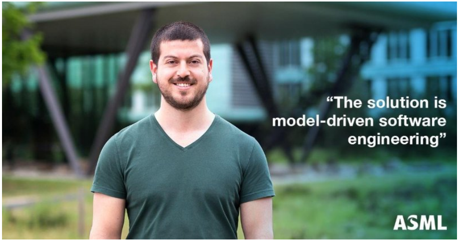
Radboud University Nijmegen
Institute for Computing and Information Sciences

- program verification using automated provers
- extended to Go channels
- *example:* concurrent prime number sieve

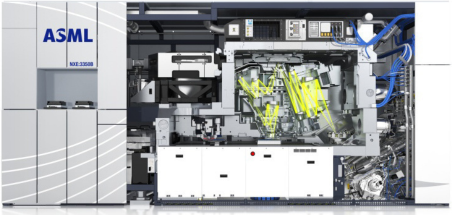
```
func sieve(src, out chan int) {
        p := <-src
        out <- p
        filt := make(chan int)
        go sieve(filt, out)
        for n := range src {
                if n%p != 0 {
                        filt <- n
                }
        }
}
```



6 Refactoring Legacy Software at ASML



- infer model of key legacy component
- use to assess quality of refactored version

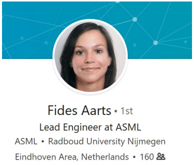


7 Why MSc Software Science at Radboud?

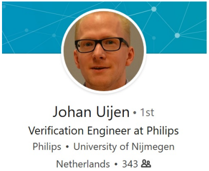
1. cutting-edge research with strong ties to industry
2. unique spectrum from theory to practice
3. very different from bachelor@RU
4. flexible thanks to electives
5. small-scale; informal atmosphere
6. excellent career prospects

7 Why MSc Software Science at Radboud?

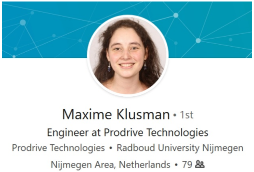
1. cutting-edge research with strong ties to industry
2. unique spectrum from theory to practice
3. very different from bachelor@RU
4. flexible thanks to electives
5. small-scale; informal atmosphere
6. excellent career prospects



Fides Aarts • 1st
Lead Engineer at ASML
ASML • Radboud University Nijmegen
Eindhoven Area, Netherlands • 160 👥



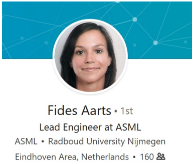
Johan Uijen • 1st
Verification Engineer at Philips
Philips • University of Nijmegen
Netherlands • 343 👥



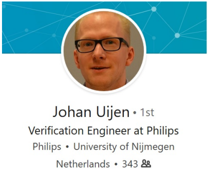
Maxime Klusman • 1st
Engineer at Prodrive Technologies
Prodrive Technologies • Radboud University Nijmegen
Nijmegen Area, Netherlands • 79 👥

7 Why MSc Software Science at Radboud?

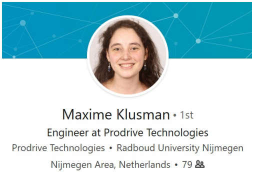
1. cutting-edge research with strong ties to industry
2. unique spectrum from theory to practice
3. very different from bachelor@RU
4. flexible thanks to electives
5. small-scale; informal atmosphere
6. excellent career prospects



Fides Aarts • 1st
Lead Engineer at ASML
ASML • Radboud University Nijmegen
Eindhoven Area, Netherlands • 160



Johan Uijen • 1st
Verification Engineer at Philips
Philips • University of Nijmegen
Netherlands • 343



Maxime Klusman • 1st
Engineer at Prodrive Technologies
Prodrive Technologies • Radboud University Nijmegen
Nijmegen Area, Netherlands • 79

Become one of tomorrow's leading software specialists!

7 Thank You! — Questions?

- *contact:*
  Frits Vaandrager <fvaan@cs.ru.nl>
  +31 24 — 36 52216
- *student advisor:*
  Perry Groot <Perry.Groot@science.ru.nl>
  +31 24 — 36 52037