

Marc van Zee

6, rue Richard Coudenhove Kalergi, Luxembourg • marcvanee@gmail.com • <http://www.marcvanzee.nl/>

Ph.D. Software Engineer in Artificial Intelligence

- PhD internship Software Engineering at Google in 2016; offers from IBM Watson and Ebay.
- Exchange to Stanford University as part of PhD thesis.
- Published 23 research papers in premier conferences, including AAAI, IJCAI, ECAI, ER, and AAMAS.
- 10+ years of programming: proficient in Java and C++.

EDUCATION

Ph.D. – Computer Science

University of Luxembourg, Luxembourg, LUX

Thesis: “Reasoning about Architecture Models”

Internship at Google Pittsburgh, USA (2016) -- Exchange to Stanford University, USA (2015)

MSc.– Artificial Intelligence, *cum laude*

Utrecht University, Utrecht, NL

Graduate Project: “Implementing Temporal Action Logics” at Linköping University, Sweden.

BSc.– Industrial Design

Eindhoven University of Technology, Eindhoven, NL

Participant in the Honours Programme

TECHNOLOGIES

Java, C, C++, Prolog, 2APL (experienced) • JavaScript, MySQL, Matlab, R (intermediate) • Python, Go, PHP, HTML/CSS (prior experience) • Coursera Machine Learning Course Certificate (Stanford University)

TOP RESEARCH PUBLICATIONS (3/18)

AGM Revision of Beliefs about Action and Time (Marc van Zee, Mehdi Dastani, Dragan Doder, Leendert van der Torre),
In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2015.

Rational Architecture = Architecture from a Recommender Perspective (Marc van Zee),
In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2015.

RationalGRL: A Framework for Rationalizing Goal Models Using Argument Diagrams (Marc van Zee, Diana Marosin, Sepideh Ghanavati, Floris Bex), *In Proceedings of the 35th International Conference on Conceptual Modeling (ER), 2016.*

A.I. INDUSTRY EXPERIENCE

Internship Software Engineering at Google, Pittsburgh (2016)

Workshop organizer of “A Decade of Collective Reasoning”, *University of Luxembourg* (2016)

Presented a tutorial on business compliance at the RE Conference, *Ottawa, Canada* (2015)

Reviewer of the Journal of Systems and Software (*J. Syst. Software*) (2015)

Board member of the A.I. association in the Benelux (*BNVKI*) (2013-2015)

Chairman of the student committee of the study association of A.I., *Utrecht University* (2010-2012)

EMPLOYMENT HISTORY

GOOGLE, 2016

PhD Internship Software Engineering

Situation: The constraint resolution algorithm for assigning attributes to products in Google Shopping has to be improved.

Task: Identify clusters of product for which the current constraint resolution algorithm performs poorly and develop an algorithm to improve it by measuring precision and recall.

Action: Investigated optimized constraint resolution approach to weighing labels from text extraction using linear programming. Created the formalism, implemented a prototype solution, and evaluated it against the existing one. C++.

Result: The implementation of the integer linear programming algorithm is part of the Google codebase and outperforms it in terms of precision and recall in various product categories. It is conceptually simpler than the previous one as well.

STANFORD UNIVERSITY, 2015

Researcher, Exchange

Situation: A collaboration with an expert in planning and metareasoning at Stanford University for Ph.D. thesis.

Task: Implement various well-known plan revision strategies and improve runtime performance.

Action: Developed an open-source testbed environment for various plan revision strategies in Java, implemented a novel dynamic plan revision strategy based on reinforcement learning.

Result: The novel dynamic plan revision strategy outperforms the existing strategies in a large number of situations.

Benchmarking results reported in the publication *Intention Reconsideration as Metareasoning* (Marc van Zee, Thomas Icard),

LINKÖPING UNIVERSITY, 2013

Master Thesis, Exchange

Situation: A temporal logic has to be implemented so a UAV (Unmanned Aerial Vehicle) can use it as a reasoning engine.

Task: Implement Temporal Action Logic (TAL) into a software tool and formally prove its correctness.

Action: Developed a general framework in Java that translates different variants of TAL and computes the results of queries using SAT/SMT solving.

Result: The first formally proven implementation of TAL into a software tool. Formal proofs and benchmarking results reported in the publication *Encoding Definitional Fragments of Temporal Action Logic into Logic Programming* (Marc van Zee, Patrick Doherty, John-Jules Meyer), In *International Workshop on Defeasible and Ampliative Reasoning (DARe)*, 2014.

UTRECHT UNIVERSITY, 2011-2013

Experimentation Project

Situation: The runtime performance for the agent programming language 2APL has to be improved

Task: Develop a caching algorithm and implement it into the source code of 2APL.

Action: Wrote a caching algorithm in Java that remembers previous queries an agent has made to the database.

Result: The runtime performance of agents using the caching algorithm is improved up to 250%, depending on the type of queries the agent performs. Benchmarking results reported in two publications:

- *Multi-Cycle Query Caching in Agent Programming* (Natasha Alechina, Tristan Behrens, Mehdi Dastani, Koen Hindriks, Koen Hubner, Fred Jomi, Brian Logan, Hai H. Nguyen, Marc van Zee), In *Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI-13)*, 2013.
- *Belief Caching in 2APL* (Mehdi Dastani, Marc van Zee), In *Workshop Engineering Multi-Agent Systems (EMAS)*, 2013.

DUTCH ASSOCIATION OF PEDIATRICS, 2010-2011

Back-end Web developer

Situation: The payment system of NVK.nl has to be digitalized.

Task: Use the Paypal API to establish a secure transaction process for digital payments that integrates into NVK.nl.

Action: Developed a workflow for digital payments using iDEAL/Paypal, integrated the workflow into the CMS.

Result: About 50 payments to the association per week (e.g., membership fees, conference registration, etc.) are carried out digitally. The administration efforts are reduced and the workflow is simplified significantly.

FABLAB AMSTERDAM, 2009

Creative Developer, Intern

Situation: A new mind-reading headset needs to be tested with a wirelessly controllable robot.

Task: Develop two actions that can be controlled with the mind connecting two actions (push and pull) to two directions of the robot (relatively forward and backward) through Flash.

Action: Coded all of the various headset software components using JavaScript, jQuery.

Result: Successfully controlled the robot via thoughts with the two new gestures. Provided successful demonstrations conferences and workshops.