



## BSR winter School

*Big Software on the Run: Where Software Meets Data*

October 23-28, 2016, Ede, the Netherlands

# Opening Wil van der Aalst

[www.3tu-bsr.nl](http://www.3tu-bsr.nl)



- **Motivation**
- **BSR project**
- **People involved**
- **Program**
- **Practicalities**



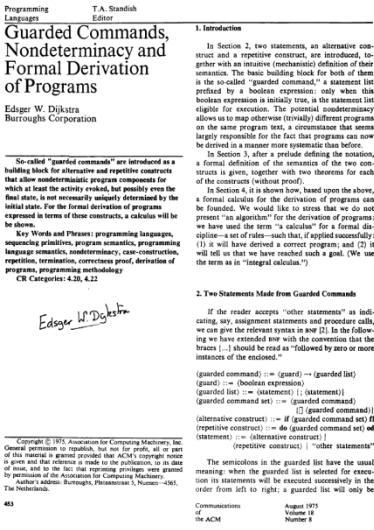
- **Motivation**
- BSR project
- People involved
- Program
- Practicalities



***in vivo  
software  
analytics***

**Big  
Software**  
on the  
**Run**

**BSR**



# Small software that did not change ...

## Guarded Commands, Nondeterminacy and Formal Derivation of Programs

Edgar W. Dijkstra  
Burroughs Corporation

So-called "guarded commands" are introduced as a building block for alternative and repetitive constructs that allow nondeterministic program components for which at least the activity evolved, but possibly even the final state, is not necessarily uniquely determined by the initial state. For the formal derivation of programs expressed in terms of these constructs, a calculus will be shown.

Key Words and Phrases: programming languages, sequencing primitives, program semantics, programming language semantics, nondeterminacy, case-construction, repetition, termination, correctness proof, derivation of programs, programming methodology  
CR Categories: 4.20, 4.22

### 1. Introduction

In Section 2, two statements, an alternative construct and a repetitive construct, are introduced, together with an intuitive (mechanistic) definition of their semantics. The basic building block for both of them is the so-called "guarded command," a statement list prefixed by a boolean expression; only when this boolean expression is initially true, is the statement list eligible for execution. The potential nondeterminacy allows us to map otherwise (trivially) different programs on the same program text, a circumstance that seems largely responsible for the fact that programs can now be derived in a manner more systematic than before.

In Section 3, after a prelude defining the notation, a formal definition of the semantics of the two constructs is given, together with two theorems for each of the constructs (without proof).

In Section 4 it is shown how, based upon the above, a formal calculus for the derivation of programs can be founded. We would like to stress that we do not present "an algorithm" for the derivation of programs; we have used the term "a calculus" for a formal discipline—a set of rules—such that, if applied successfully: (1) it will have derived a correct program; and (2) it will tell us that we have reached such a goal. (We use the term as in "integral calculus.")

### 2. Two Statements Made from Guarded Commands

If the reader accepts "other statements" as indicating, say, assignment statements and procedural calls, we can give the relevant syntax in now (2). In the following we have extended now with the convention that the braces  $\{ \}$  should be read as "followed by zero or more instances of the enclosed."

(guarded command) ::= (guard) → (guarded list)  
(guard) ::= (boolean expression)  
(guarded list) ::= (statement) ; (statement)  
(guarded command set) ::= (guarded command) | (guarded command)  
(alternative construct) ::= if (guarded command set) fi  
(repetitive construct) ::= do (guarded command set) od  
(statement) ::= (other statement) | (guarded command)

The notations in the guarded list have the usual meaning: when the guarded list is selected for execution its statements will be executed successively in the order from left to right; a guarded list will only be

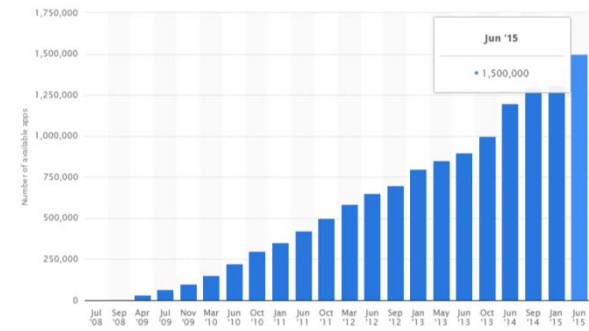
Copyright © 1975, Association for Computing Machinery, Inc. General permission to reproduce, but not for profit, all or part of this material is granted provided that ACM's copyright notice is given and that the abstract is made to the publisher, or its date of issue, and to the fact that the reproduction was granted by permission of the Association for Computing Machinery. Author's address: Burroughs, Philadelphia 1, Newnes—404, The Netherlands.

Communications  
of the ACM  
August 1975  
Volume 18  
Number 8

$\{ \text{var } x, y: \text{int}; \{ X_{gcd} Y = x \wedge y > 0 \} \}$   
 $\{ \text{do } x > y \rightarrow x := x - y \}$   
 $\{ \text{od } \{ X_{gcd} Y = x \wedge X_{gcd} Y = y \} \}$

$\text{fib}_0 = 0$   $\text{fib}_1 = 1$   $\text{fib}_{(n+2)} = \text{fib}_{(n+1)} + \text{fib}_n$   
 $\text{fib}(X_{gcd} Y)$   
 $\text{gcd fib}(X_{gcd} Y)$   
 $\text{fib}(y)$

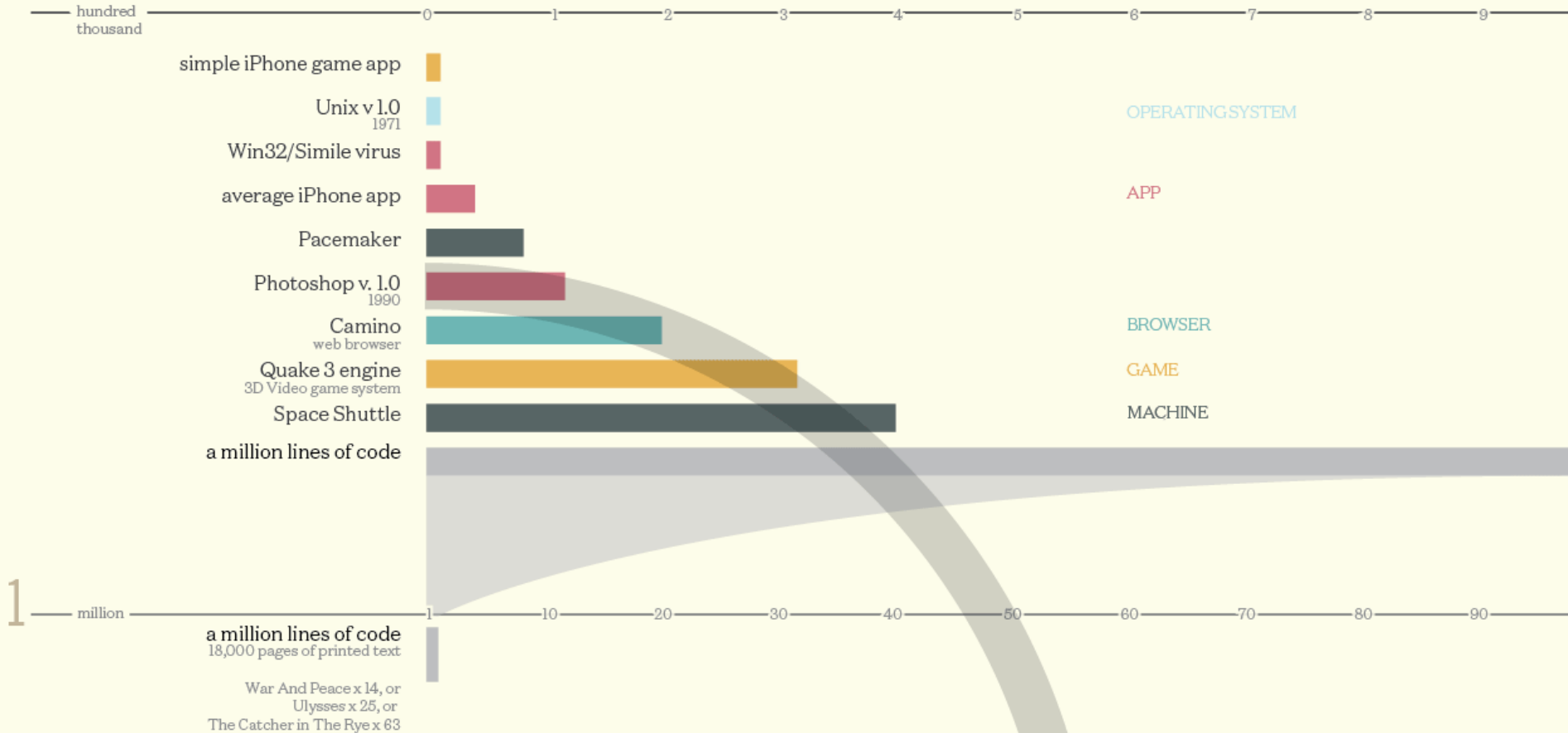
$\text{fib}(X_{gcd} Y) = (\text{fib}.x) \text{gcd}(\text{fib}.y)$   
 $\text{fib}(a+b) \text{gcd fib}$   
 $\text{fib}(a+b)$   
 $\{ \text{FIBONACCI} \}$   
 $f(a-1) \cdot \text{fib} + \text{fib}$



**Big software on the run ...**

# Codebases

Millions of lines of code



50

Windows 7  
2009Windows XP  
2001

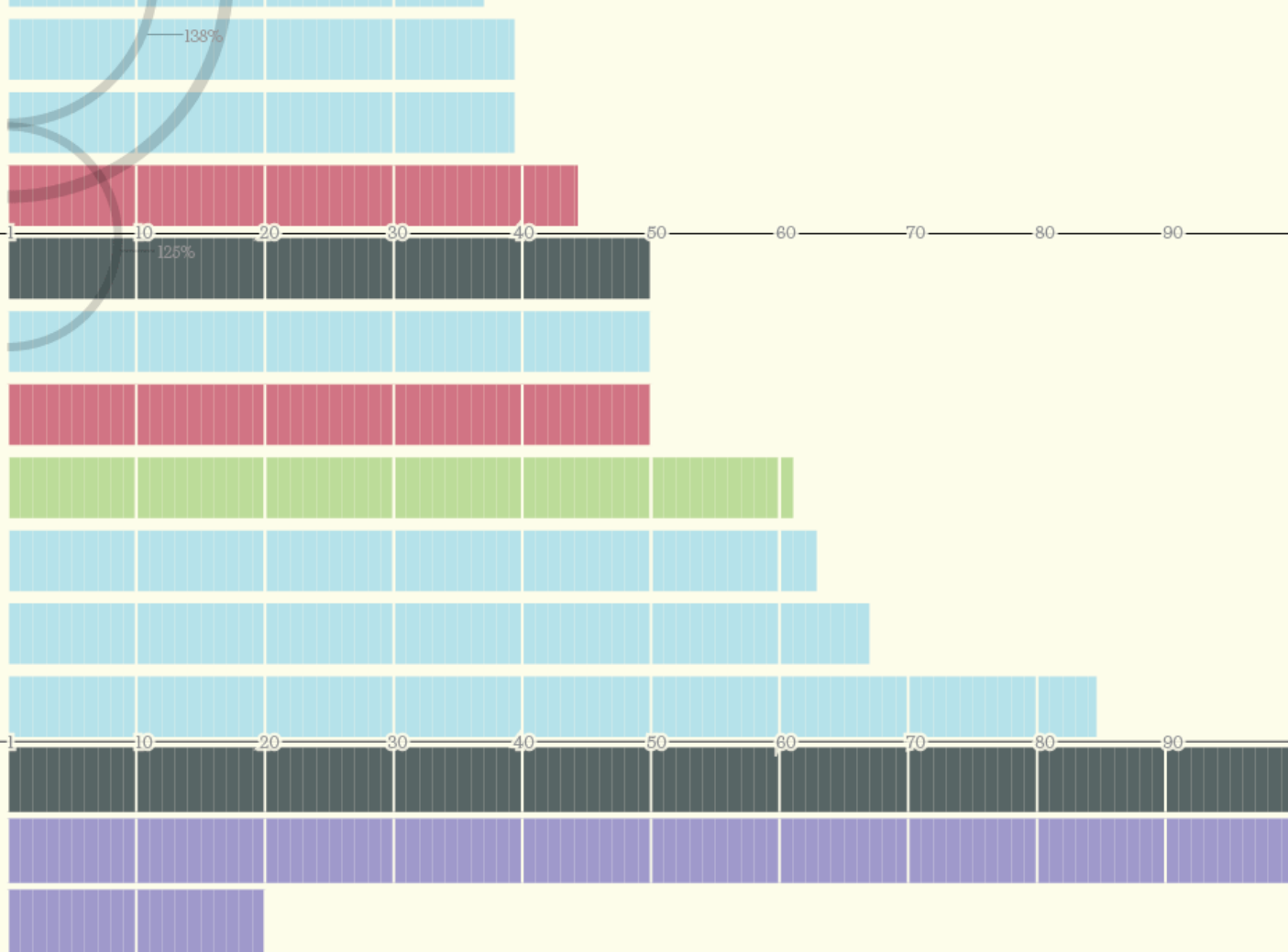
Microsoft Office 2013

Large Hadron Collider  
total codeWindows Vista  
2007

Microsoft Visual Studio 2012

Facebook  
(including backend code)US Army Future Combat System  
fast battlefield network system (aborted)Debian 5.0 codebase  
free, open-source operating systemMac OS X "Tiger"  
v10.4Car software  
average modern high-end carMouse\*  
Total DNA basepairs in genome

100

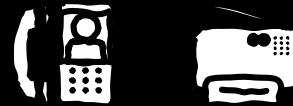
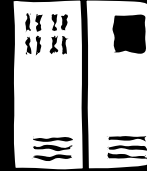


**Big**  
*software*  
is like a  
**Living**  
*organism*

growing scale

growing complexity

continuous evolution  
of software



continuously changing  
environment



increasing diversity

threats to security and  
thrust

# Big software is like a *Living organism*

not-standalone  
interconnected  
distributed

cloud  
Hadoop  
sensor

late composition  
updates  
apps

multiple platforms  
many versions  
many configurations

hardware changes  
OS, network, etc.  
unanticipated use

continuously changing  
environment

we rely on  
software in a  
hostile open world

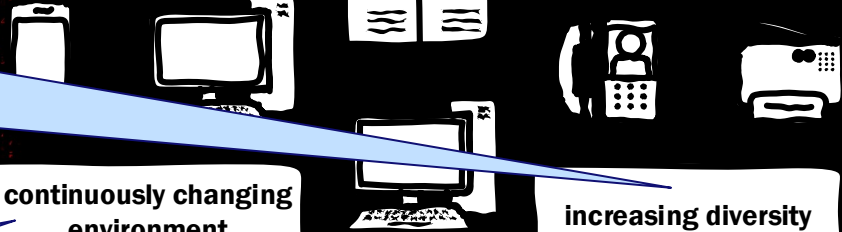
growing complexity

continuous evolution  
of software

increasing diversity

threats to security and  
thrust

growing scale





- Motivation
- **BSR project**
- People involved
- Program
- Practicalities

*in vivo*  
**software  
analytics**

**Big  
Software  
on the  
Run**

BSR

~~€34,722,000~~

€1,999,989



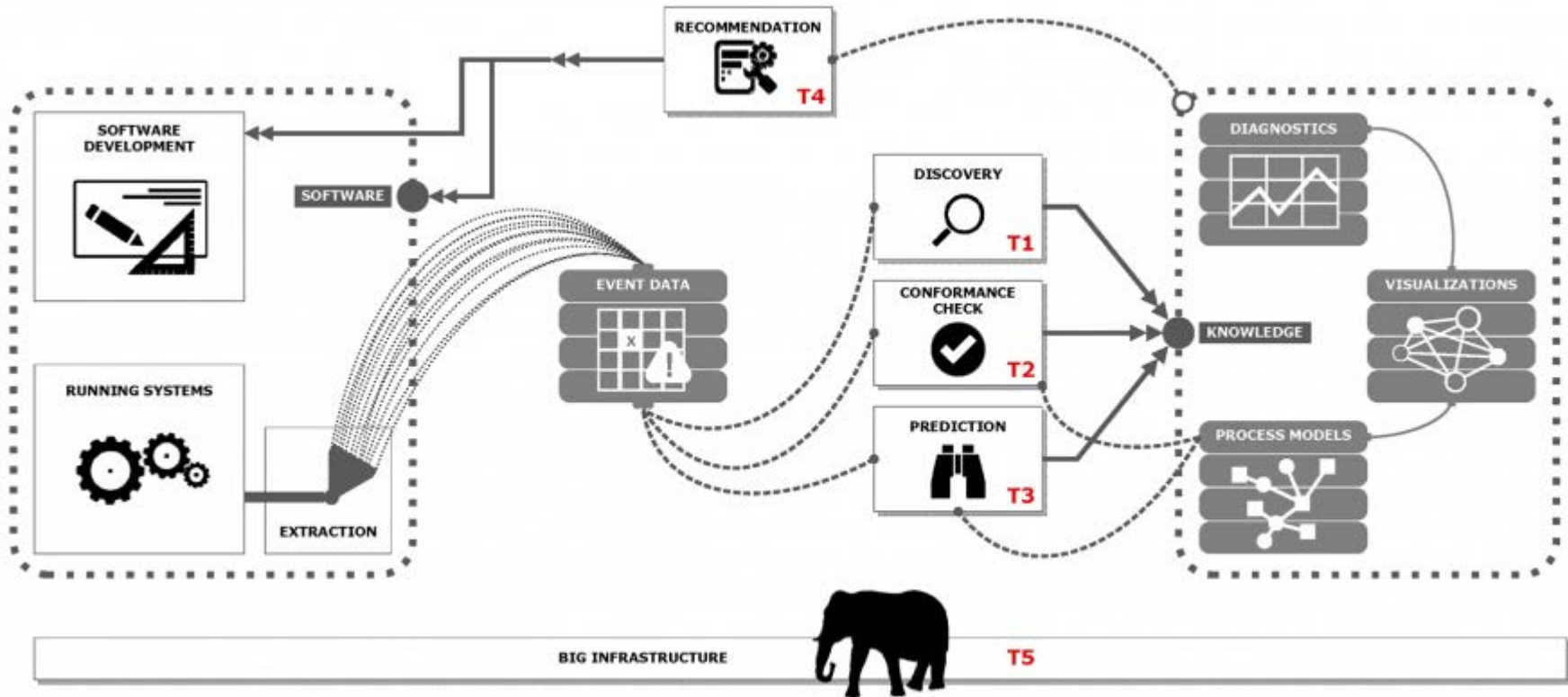
**Wil van der Aalst &  
Jack van Wijk**



**Arie van Deursen &  
Inald Legendijk**

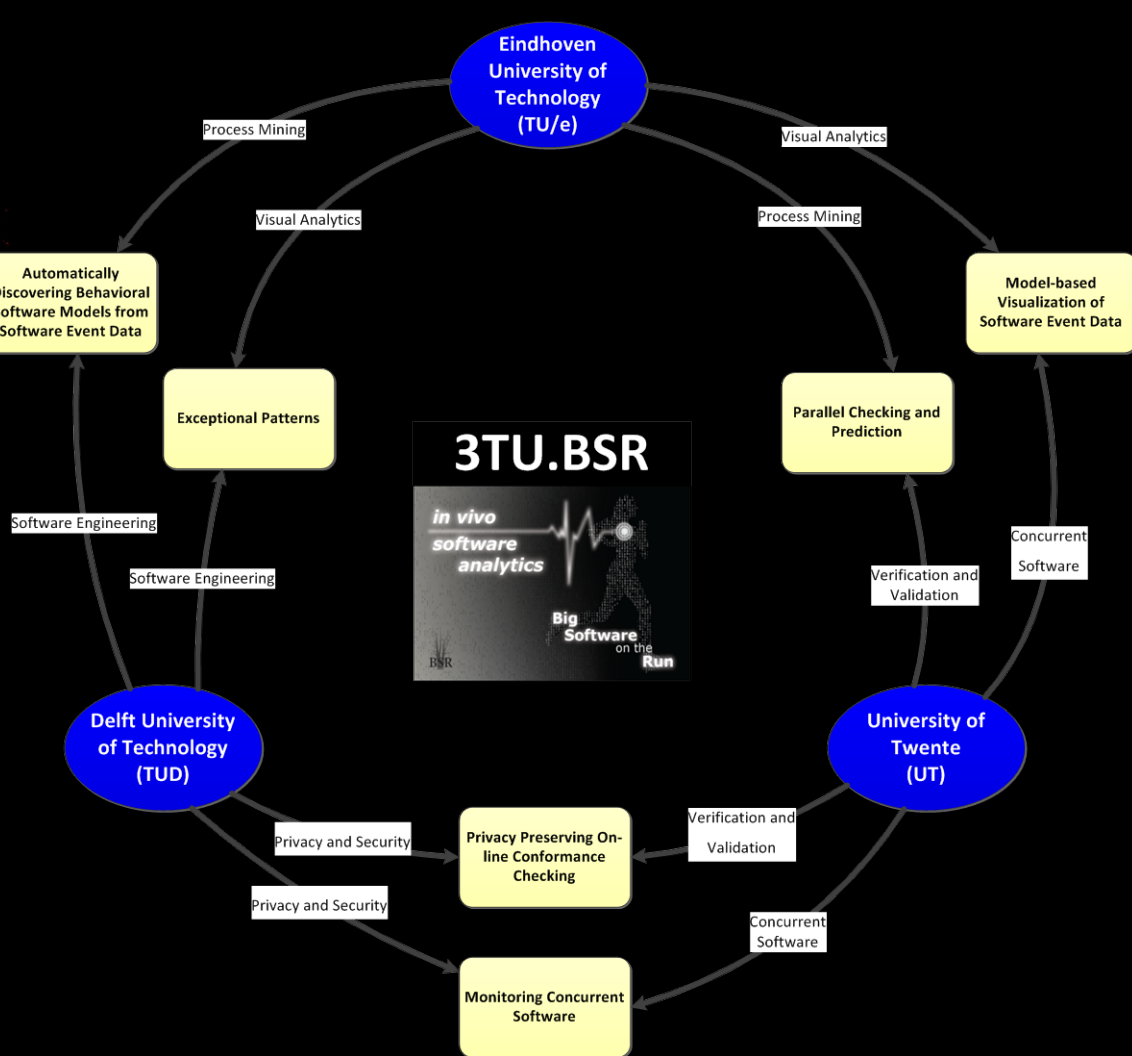
**UNIVERSITY OF TWENTE.**

**Jaco van de Pol &  
Marieke Huisman**



**Big**  
software  
is like a  
**Living**  
organism

6 PhDs + 3 postdocs  
2015-2019



# BSR Team



UNIVERSITY OF TWENTE.



**Annibale Panichella, Arie van Deursen, Arnd Hartmanns, Bart Postma, Boudewijn van Dongen, Cong Liu, Gamze Tillem, Huub van de Wetering, Inald Lagendijk, Jack van Wijk, Jaco van der Pol, Mariëlle Stoelinga, Maikel Leemans, Marieke Huisman, Nour Assy, Sicco Verwer, Vincent Bloemen, Wil van der Aalst, Zekeriya Erkin, Moritz Beller, Freark van den Berg, and Mozhan Soltani.**

# *in vivo software analytics*

visit [www.3tu-bsr.nl](http://www.3tu-bsr.nl)

**Big  
Software  
on the  
Run**



BSR



- Motivation
- BSR project

- **People involved**

- Program
- Practicalities

# Organization of this BSR Winter School

## School directors



Wil van der Aalst



Arie van Deursen



**Main organizer:  
Nour Assy!**

# Scientific board



Jaco van der Pol



Jack van Wijk



Boudewijn van Dongen



Marieke Huisman



Inald Lagendijk



Marielle Stoelinga



Annibale Panichella



Arnd Hartmaans

# Organizing committee



Nour Assy



Ine van der Ligt



Tamara Brusik



Moritz Beller



Bart Postma



Cong Liu



Gamze Tillem



Maikel Leemans



Vincent Bloemen



- Motivation
- BSR project
- People involved
- **Program**
- Practicalities

Sunday 23		Monday 24	Tuesday 25	Wednesday 26	Thursday 27	Friday 28
	8:45 - 9:00	Opening session Wil van der Aalst				
	9:00 - 10:30	Wil van der Aalst Lecture 1	Margaret-Anne Storey Lecture 3	Robert DeLine Lecture 6	Georgios Gousios Lecture 8	Bram Adams Lecture 11
		poster presentation (2min/participant)	Coffee break + posters			
	11:00 - 12:30	Poster session + drinks	Frits Vaandrager Lecture 4	Mark van den Brand Lecture 7 Best poster presentation (12:00 - 12:30)	Jaco van de Pol Lecture 9	Zekeriya Erkin Lecture 12
		Lunch				Closing session Arie van Deursen (12:30 - 13:00)
	14:00 - 15:30	Jack van Wijk Lecture 2	Arie van Deursen Lecture 5	Excursion (departure at 13:30)	Marieke Huisman Lecture 10	Lunch
15:00 - 17:00 Registration & checking-in			Coffee break + posters	Excursion		
17:00 - 18:00 Welcome reception + drinks	16:00 - 18:00	Hands-on session (Tu/e)	Hands-on session (TUD)	Excursion	Hands-on session (UT)	
		Dinner 20:00 - 22:00				

 **Margaret-Anne Storey, University of Victoria, British Columbia**



*Dr. Margaret-Anne Storey is a Professor of Computer Science at the University of Victoria, a Canada Research Chair in Human and Social Aspects of Software Engineering, and a Principal Investigator for the Canadian Centre for Software Engineering Research (CSER). Her research goal is to understand how technology can help people explore, understand and share complex information and knowledge. She evaluates and applies techniques from knowledge engineering, social software and visual interface design to applications such as collaborative software development, program comprehension, biomedical ontology development, and learning in Web-based environments. Some of her past projects include investigating the role of social media in collaborative software engineering, improving information visualization techniques and developing social software to facilitate the next version of the International Classification of Diseases with the World Health Organization. Dr. Storey recently wrapped up a 10-year project as a Principal Investigator for the National Center for Biomedical Ontology with the National Institutes of Health in the United States.*

 **Robert DeLine, Microsoft research, USA**



*Dr. Robert DeLine is a principal researcher at Microsoft Research, working at the intersection of software engineering and human-computer interaction. His research group designs development tools in a user-centered fashion: they conduct studies of development teams to understand their work practice and prototype tools to improve that practice. Dr. DeLine has a background in both HCI and software engineering. His master's thesis was the first version of the Alice programming environment (University of Virginia, 1993), and his PhD was in software architecture (Carnegie Mellon University, 1999).*

 **Bram Adams, Polytechnique Montreal, Canada**



*Dr. Bram Adams is an assistant professor at the École Polytechnique de Montréal, where he heads the MCIS lab on Maintenance, Construction and Intelligence of Software. He obtained his PhD at Ghent University (Belgium), and was a postdoctoral fellow at Queen's University (Canada) from October 2008 to December 2011. His research interests include software release engineering in general, and software integration, software build systems, software modularity and software maintenance in particular. His work has been published at premier venues like ICSE, FSE, ASE, ESEM, MSR and ICSM, as well as in major journals like EMSE, JSS and SCP.*

 **Georgios Gousios, Delft University of Technology, the Netherlands**




*Dr. Georgios Gousios is an assistant professor at the Web Information Systems group, Delft University of Technology. His research interests include software engineering, software analytics and programming languages. He works in the fields of distributed software development processes, software quality, software testing, developer productivity assessment, research infrastructures and software security. His research has been published in top venues (ICSE, FSE, EMSE, MSR) where he has received 4 best paper awards. In total, he has published more than 50 papers and also co-edited the "Beautiful Architectures" book (O'Reilly, 2009). He is the main author of the GHTorrent data collection and curation framework and the Altheia Core repository mining platform. Dr. Gousios holds a PhD in Empirical Software Engineering (mining software repositories) with distinction from the Athens University of Economics and Business (AUEB) and an MSc (software engineering) with distinction from the University of Manchester.*

 Margaret-Anne Storey



Dr. Ma Aspect research evaluation collaborate Some visualization techniques a Organization. Dr. Storey r Institutes of Health in the U

 Frits Vaandrager, Radboud University, the Netherlands



Prof. Frits Vaandrager is a full professor of Informatics for Technical Applications, Institute for Computing and Information Sciences. He has a strong interest in the development and application of theory, (formal) methods and tools for the specification and analysis of computer based systems. In particular, he is interested in real-time embedded systems, distributed algorithms and protocols. Together with Lynch, Segala, and Kaynar he developed the (timed, probabilistic and hybrid) input/output automata formalisms, which are basic mathematical frameworks to support description and analysis of computing systems. He has been and is involved in a large number of projects in which formal verification and model checking technology is applied to tackle practical problems from industrial partners. Recently, he has also become interested in automata learning.

 Robert DeLine, Micros



Dr. Ro interact unders engine softwa

 Mark van den Brand, Eindhoven University of Technology, the Netherlands




Prof. Mark van den Brand started his study computer science in 1982 at the Radboud University Nijmegen (The Netherlands). In 1987 he became a PhD student at the Radboud University Nijmegen. In 1992 he started as assistant professor in the Programming Research Group at the University of Amsterdam. In 1997 he switched from University of Amsterdam to CWI (Centrum voor Wiskunde en Informatica) and became senior researcher and project leader on the ASF+SDF project. The ASF+SDF project is one of the main research topics of the Interactive Software Development and Renovation group. Since 2006 Mark van den Brand is a full professor of Software Engineering and Technology at TU/e in the Department of Mathematics and Computer Science. Since May 2009 he is visiting professor at Royal Holloway, University of London. His current research activities are on generic language technology, and model driven engineering. A number of his research topics are to investigate the correctness and quality aspects of model transformations, and the use of meta-modeling techniques in the area of functional safety. The SET group participates/participated a number of (industrial) projects related to model driven engineering and software evolution. The goal of his research is to investigate the correctness and quality aspects of model transformations. He is president of the European Association of Programming Languages and Systems (EAPLS) and chair of the steering committee of Software Language Engineering. He was six times guest editor of special issues of Science of Computer Programming on academic software development (Experimental Software and Toolkits (EST)). He is member of the editorial board of the journal of Science of Computer Programming, Central European Journal for Computer Science, and Journal of Computer Languages, Systems and Structures.

 Bram Adams, Polytec



Dr. Br Constr Univer softwa premie

 Wil van der Aalst, Eindhoven University of Technology



Prof. Wil van der Aalst is a full professor of Information Systems at the Technische Universiteit Eindhoven (TU/e). At TU/e he is the scientific director of the Data Science Center Eindhoven (DSC/e). Since 2003 he holds a part-time position at Queensland University of Technology (QUT). His personal research interests include workflow management, process mining, Petri nets, business process management, process modeling, and process analysis. Wil van der Aalst has published more than 180 journal papers, 18 books (as author or editor), 400 refereed conference/workshop publications, and 60 book chapters. Many of his papers are highly cited (he one of the most cited computer scientists in the world and has an H-index of 122 according to Google Scholar) and his ideas have influenced researchers, software developers, and standardization committees working on process support. He has been a co-chair of many conferences including the Business Process Management conference, the International Conference on Cooperative Information Systems, the International conference on the Application and Theory of Petri Nets, and the IEEE International Conference on Services Computing. He is also editor/member of the editorial board of several journals, including Computing, Distributed and Parallel Databases, Software and Systems Modeling, the International Journal of Business Process Integration and Management, the International Journal on

 Georgios Gousios, De



Dr. Ge include proces has be papers currat software repositories) with University of Manchester.

🌐 Margaret-Anne Storey, RA



Dr. Ma Aspect research evaluation collaboration. Some visualization techniques at Organization. Dr. Storey r Institutes of Health in the U

🌐 Frits Vaandrager, RA



Prof. a str base Segn fram form become interested in au

🌐 Arie van Deursen, Delft University of Technology



Prof. Arie van Deursen is a full professor at Delft University of Technology, where he is heading the Software Engineering Research Group. arie's research is concerned with the question how people build and evolve software systems, what the problems are with existing software systems, and whether there are better ways of doing software development and evolution. Arie has published in the areas of software testing, software architecture, web 2.0 (and Ajax in particular), aspect-oriented programming, reverse engineering, repository mining, program comprehension, and model-driven engineering. Arie is on the editorial board of Springer's Empirical Software Engineering, Springer's Computing, and of Wiley's Journal of Software Maintenance and Evolution: Research and Practice. He also serves as a steering committee member of the International Conference on Software Maintenance (ICSM).

🌐 Jaco van de Pol, University of Twente



Prof. Jaco van de Pol is a full professor at University of Twente, where he is heading the Formal Methods and Tools group. His research interest include modeling and analysis of safety, dependability and security aspects of software-intensive embedded systems by means of model checking, theorem proving and testing. More specifically, the development of new model checking techniques based on symbolic techniques (e.g. abstraction, confluence, fixpoint equation systems, constraint solving) and high-performance computing (e.g. distributed model checking, multi-core and out-of-core algorithms). Application domains include embedded systems, distributed systems, security protocols, and biological systems.

🌐 Zekeriya Erkin, Delft University of Technology



Dr. Zekeriya Erkin is an assistant professor in the Cyber Security Group, where he works on Privacy Enhancing Technologies. His main research goal is to protect private data in online applications against misuse. One way to achieve this goal is to hide the private data by means of encryption in such a way that without the authorization of the data owner, no one can see the content but still process the encrypted data for the required service. One simple example is to encrypt an image and ask the service provider if there is a face image of a suspect. Since the image is encrypted, there is no way of saying for the service provider. However, it is possible to determine this by running interactive, cryptographic protocols based on Homomorphic Encryption and Secure Multiparty Computation techniques.

🌐 Jack van Wijk, Eindhoven University of Technology



Prof. Jack van Wijk is a full professor in visualization at the Department of Mathematics and Computer Science of Eindhoven University of Technology (TU/e). He received a MSc degree in industrial design engineering in 1982 and a PhD degree in computer science in 1986, both from Delft University of Technology, both with honors. After a short period in the software industry, he has worked for ten years at the Netherlands Energy Research Foundation ECN. He joined Eindhoven University of Technology in 1998, where he became a full professor of visualization in 2001. His main research interests are information visualization, visual analytics, and mathematical visualization. He is cofounder and VP Scientific Affairs of MagnaView BV. He has (co-)authored more than 150 papers in visualization and computer graphics.

He has been paper cochair for IEEE Visualization (2003, 2004), IEEE InfoVis (2006, 2007), IEEE VAST 2009, IEEE PacificVis 2010 and EG/IEEE EuroVis 2011. He received the IEEE Visualization Technical Achievement Award in 2007 for his work on flow visualization; the Eurographics 2013 Outstanding Technical Contributions Award; best paper awards at IEEE InfoVis 2003, IEEE Visualization 2005, IEEE PacificVis 2013, IEEE InfoVis 2014; IEEE VAST 2015, and the 2009 Henry Johns Award of The Cartographic Journal. He was keynote, capstone, or invited speaker at PacificVis 2009, EuroVAST 2010, GD 2011, VMV 2011, i-KNOW 2011, Bridges 2013, IEEE VISSOFT 2013, IEEE VIS 2013 (video), and VAHC 2015.

🌐 Robert DeLine, Microsoft



Dr. Rob interac unders engine softwa

🌐 Mark van den Brand



Prof. beca at th beca Inter Tech

University of London. Hi investigate the correctness group participates/parti investigate the correctness (EAPLS) and chair of th Programming on academi Computer Programming

🌐 Bram Adams, Polytechnic



Dr. Br Constr Univer softwa premie

🌐 Georgios Gousios, De



Dr. Ge include proces has be papers currat software repositories) with University of Manchester.

🌐 Wil van der Aalst, Eindhoven University of Technology



Prof. direc (QU) mod conf in th

standardization commit the International Confere International Conference Parallel Databases, Soft

• Margaret-Anne Storey, RA



Dr. Ma Aspect research evaluation collaborative. Some visualization techniques a Organization. Dr. Storey r Institutes of Health in the U

• Frits Vaandrager, RA



Prof. a str base Segn fram form become interested in au

• Arie van Deursen, Delft University of Technology



committee member of the International Software Maintenance (ISM)

Prof. Arie van Deursen is a full professor at Delft University of Technology, where he is heading the Software Engineering Research Group. Arie's research is concerned with the question how people build and evolve software systems, what the problems are with existing software systems, and whether there are better ways of doing software development and evolution. Arie has published in the areas of software testing, software architecture, web 2.0 (and Ajax in particular), aspect-oriented programming, reverse engineering, repository mining, program comprehension, and model-driven engineering. Arie is on the editorial board of Springer's Empirical Software Engineering, Springer's Computing, and of Wiley's Journal of Software Maintenance and Evolution: Research and Practice. He also serves as a steering

• Marieke Huisman, University of Twente



Prof. Marieke Huisman is a professor in the FMT (Formal Methods and Tools) Group at the University of Twente working on reliability and correctness of concurrent and distributed software. She is the winner of the Dutch ICT award 2013. She has a strong track record on analysis of concurrent software. In her ERC project VerCors, she developed tools for static analysis of concurrent programs, using different concurrency paradigms (shared memory, but also vector programs).

• Jaco van



• Robert DeLine, Microsoft



Dr. Rob interact unders engine softwa

• Mark van den Brand



Prof. beca at th beca Inter Tech

University of London. Hi investigate the correctness group participates/parti investigate the correctness (EAPLS) and chair of th Programming on academi Computer Programming

• Bram Adams, Polytec



Dr. Br Constr Univer softwa premie

• Zekeriya Erkin, Delft University of Technology



Dr. Zekeriya Erkin is an assistant professor in the Cyber Security Group, where he works on Privacy Enhancing Technologies. His main research goal is to protect private data in online applications against misuse. One way to achieve this goal is to hide the private data by means of encryption in such a way that without the authorization of the data owner, no one can see the content but still process the encrypted data for the required service. One simple example is to encrypt an image and ask the service provider if there is a face image of a suspect. Since the image is encrypted, there is no way of saying for the service provider. However, it is possible to determine this by running interactive, cryptographic protocols based on Homomorphic Encryption and Secure Multiparty Computation techniques.

• Jack van Wijk, Eindhoven University of Technology



Prof. Jack van Wijk is a full professor in visualization at the Department of Mathematics and Computer Science of Eindhoven University of Technology (TU/e). He received a MSc degree in industrial design engineering in 1982 and a PhD degree in computer science in 1986, both from Delft University of Technology, both with honors. After a short period in the software industry, he has worked for ten years at the Netherlands Energy Research Foundation ECN. He joined Eindhoven University of Technology in 1998, where he became a full professor of visualization in 2001. His main research interests are information visualization, visual analytics, and mathematical visualization. He is cofounder and VP Scientific Affairs of MagnaView BV. He has (co-)authored more than 150 papers in visualization and computer graphics.

He has been paper cochair for IEEE Visualization (2003, 2004), IEEE InfoVis (2006, 2007), IEEE VAST 2009, IEEE PacificVis 2010 and EG/IEEE EuroVis 2011. He received the IEEE Visualization Technical Achievement Award in 2007 for his work on flow visualization; the Eurographics 2013 Outstanding Technical Contributions Award; best paper awards at IEEE InfoVis 2003, IEEE Visualization 2005, IEEE PacificVis 2013, IEEE InfoVis 2014; IEEE VAST 2015, and the 2009 Henry Johns Award of The Cartographic Journal. He was keynote, capstone, or invited speaker at PacificVis 2009, EuroVAST 2010, GD 2011, VMV 2011, i-KNOW 2011, Bridges 2013, IEEE VIS 2013 (video), and VAHC 2015.

• Georgios Gousios, De



Dr. Ge include proces has be papers currat software repositories) with University of Manchester.

• Wil van der Aalst, Eindhoven University of Technology



Prof. direc (QU) mod conf in th

standardization commit the International Confer International Conference Parallel Databases, Soft



- Motivation
- BSR project
- People involved
- Program

- **Practicalities**

Sunday 23	Monday 24	Tuesday 25	Wednesday 26	Thursday 27	Friday 28
	8:45 - 9:00 Opening session Wil van der Aalst				
	9:00 - 10:30 Wil van der Aalst Lecture 1	Margaret-Anne Storey Lecture 3	Robert DeLine Lecture 6	Georgios Gousios Lecture 8	Bram Adams Lecture 11
	poster presentation (2min/participant)	Coffee break + posters			
					Triya Erkin Lecture 12
					Opening session van Deursen (12:30 - )
	14:00 - 15:30 Jack van Wijk Lecture 2	Arie van Deursen Lecture 5	Excursion (departure at 13:30)	Marieke Huisman Lecture 10	Lunch
15:00 - 17:00 Registration & checking-in		Coffee break + posters	Excursion		
17:00 - 18:00 Welcome reception + drinks	16:00 - 18:00 Hands-on session (Tu/e)	Hands-on session (TUD)	Excursion	Hands-on session (UT)	
		Dinner 20:00 - 22:00			

# Session chairs

- **Monday: Wil van der Aalst**
- **Tuesday: Arie van Deursen**
- **Wednesday: Inald Lagendijk**
- **Thursday: Jaco van der Pol**
- **Friday: Zekeriya Erkin**

Sunday 23	Monday 24	Tuesday 25	Wednesday 26	Thursday 27	Friday 28
	8:45 - 9:00 Opening session Wil van der Aalst				
	9:00 - 10:30 Wil van der Aalst Lecture 1	Margaret-Anne Storey Lecture 3	Robert DeLine Lecture 6	Georgios Gousios Lecture 8	Bram Adams Lecture 11
	poster presentation (2min/participant)	Coffee break + posters			
	11:00 - 12:30 Poster session + drinks	Frits Vaandrager Lecture 4	Mark van den Brand Lecture 7 Best poster presentation (12:00 - 12:30)	Jaco van de Pol Lecture 9	Zekeriya Erkin Lecture 12
		Lunch			Closing session Arie van Deursen (12:30 - 13:00)
	14:00 - 15:30 Jack van Wijk Lecture 2	Arie van Deursen Lecture 5	Excursion (departure at 13:30)	Marieke Huisman Lecture 10	Lunch
15:00 - 17:00 Registration & checking in		Coffee break + posters	Excursion		
17:00 - 18:00 Welcome reception + drinks	16:00 - 18:00 Hands-on session (TU/e)	Hands-on session (TU/e)	Excursion	Hands-on session (UT)	
		Dinner 20:00 - 22:00			

# Poster sessions

- 2 minute presentations (strict)
- First session this morning

Monday 24		T
8:45 - 9:00	Opening session Wil van der Aalst	
9:00 - 10:30	Wil van der Aalst Lecture 1	M S L
	poster presentation (2min/participant)	C p
11:00 - 12:30	Poster session + drinks	F L
		L
14:00 - 15:30	Jack van Wijk Lecture 2	A L
		C P
16:00 - 18:00	Hands-on session (Tu/e)	H (
		D

# Three hands-on sessions

- **Monday, Tuesday and Thursday**
- **Descriptions and materials are now available at <http://www.3tu-bsr.nl/doku.php?id=bsr-ws2016:handson>.**
- **In each session, a mini-challenge is organized and the winner will be announced during the dinner of the next day!**
- **Please make sure to prepare your laptops and install the required tools if applicable.**

posters			
16:00 - 18:00	Hands-on session (Tu/e)	Hands-on session (TUD)	Excursion
			Hands-on session (UT)
Dinner 20:00 - 22:00			

# Excursion

(Wednesday afternoon)

Kröller  
Müller





## BSR winter School

*Big Software on the Run: Where Software Meets Data*

October 23-28, 2016, Ede, the Netherlands



enjoy