UML-Driven Software Performance Engineering:

A systematic mapping and a review of several selected techniques

Doç. Dr. Vahid Garousi
Associate Professor of Software Engineering
Senior Software Consultant

Sistem ve Yazılım Kalite Mühendisliği Araştırma Grubu (SySoQual)
Yazılım Mühendisliği Bölümü
Atılım Üniversitesi

www.atilim.edu.tr/~vahid.garousi
vahid.garousi@atilim.edu.tr

Yazılım Danışmanı
Maral Yazılım Mühendislik Hizmetleri
Ankara

1st International Workshop on Advanced Topics on Software Engineering
ATSEN 2014
İKÜ
7 November 2014

Outline

- **Background of the speaker and his research expertise**
- **A systematic mapping of UML-Driven Software Performance Engineering (UML-SPE)**
- **Review of one UML-SPE technique**

Background of the speaker and his research background

- **Education:**
  - Professional Engineer designation (P.Eng.), Province of Alberta, Canada, 2008
  - PhD (Carleton University, Canada), 2006
  - MSc (University of Waterloo, Canada), 2003
  - BSc (Sharif University of Technology, Iran), 2000
- **Work Experience:**
  - Atilim University, Ankara, Türkiye, since Jan. 2014
  - Middle East Technical University, Ankara, Türkiye, Jan.-Dec. 2013
    - Visiting Associate Professor, Graduate School of Informatics
  - University of Calgary, Canada
    - Associate Professor of Software Engineering, April 2010-2014 (tenured)
    - Assistant Professor of Software Engineering, 2006-2011
- **Research Expertise:**
  - Software Engineering
  - Software Testing and Quality Assurance (QA)
  - Software Requirements and Software Maintenance
  - "Action research", i.e., industry-academia collaborations
- **Personal:**
  - Canadian citizen, of Azerbaijani-Iranian origin

Software Quality Engineering Research Group (SoftQual), 2006-2014

Software Quality Engineering Research Group

Engineering Quality Software

University of Calgary, Alberta, Canada
www.softqual.ucalgary.ca

- **www.softqual.ucalgary.ca**

- **Alumni**
  - **1 Post-doctoral fellow**
  - **3 PhD students**
  - **9 MSc**
  - **30+ undergraduate students**

- **Almost all our projects were applied R&D projects in collaboration with the industry. More than 10 industrial partners such as IBM**
- **More than $1.5 Million CND in funding in 7 years**
- **Output: More than 29 journal papers and 35+ conference papers**

Software Quality Engineering Research Group (SoftQual) - Research Projects

Research Projects and Industrial Partners

Project Title	Duration	Funded by...	In collaboration with...
Improving the cost effectiveness of software testing activities and processes	2012-2013	NSERC ENGAGE grant #EGP 444884-12	Pason Systems Corp.
Automated software testing of communication frameworks	2012-2013	NSERC ENGAGE grant #EGP 437020-12	Telvent
Tuning of Artifact and Process Parameters towards Optimized Maintenance	2012-2013	NSERC CRD grant #CRDPJ 414157-11 and industry	NovAtel Inc. A joint project with the Software Engineering Decision Support Research Laboratory
Mining Repositories for Optimized Embedded Software Life-cycle Effort Allocation	2011 (May-October)	NSERC ENGAGE grant #EGP 413039-11	NovAtel Inc. A joint project with the Software Engineering Decision Support Research Laboratory
Design and Testing of defect-intolerant Embedded Systems. A multi-disciplinary team. Our team's focus area in this group project: • Papers: 2010, 2011 • Open-source tools developed for this	2008-2011	NSERC CRD grant #CRDPJ 365295-08 and industry	Analog Devices Inc. CDL Systems Ltd. Ecole d'ingénieurs et d'architectes de Fribourg DirectVoxx

System and Software Quality Engineering Research Group (SySoQual), since 2014

SySoQual
System and Software Quality Engineering Research Group
Atilim University, Ankara, Turkey
www.se.atilim.edu.tr

SySoQual
Sistem ve Yazılım Kalite Mühendisliği Araştırma Grubu
Atılım Üniversitesi, Ankara, Türkiye
www.se.atilim.edu.tr/sysoqual

- se.atilim.edu.tr/sysoqual
- Students:
  - 3 MSc students
  - 10+ undergraduate students
- Almost all our projects have been applied R&D projects in collaboration with the industry.
- Has provided several consulting R&D projects
- Already involved in one ITEA2 and several TUBITAK grants
- In process of applying for more H2020, ITEA3 and TUBITAK grants
- Some of industry partners:

arçelik

UNIT INFORMATION TECHNOLOGIES

T.C. BAŞBAKANLIK YURTDIŞI TÜRKLER VE AKRABA TOPLULUKLAR BAŞKANLIĞI

ARDGRUP Bilişim | Medikal | Danışmanlık

intest inovatif test çözümleri

Simsoft "İNSAN İÇİN İNSAN MERKEZLİ ÇÖZÜMLER"

Outline

- **Background of the speaker and his research expertise**
- **A systematic mapping of UML-Driven Software Performance Engineering (UML-SPE)**
- **Review of one UML-SPE technique**

A recent book chapter…

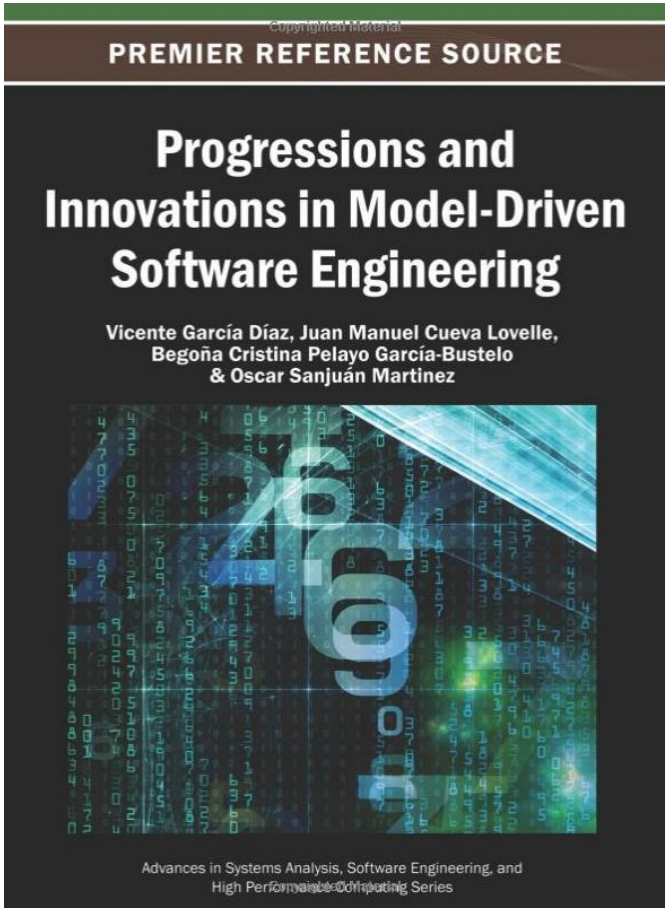
Chapter 2

UML-Driven Software Performance Engineering: A Systematic Mapping and Trend Analysis

Vahid Garousi
University of Calgary, Canada

Shawn Shahnewaz
University of Calgary, Canada

Diwakar Krishnamurthy
University of Calgary, Canada



Progressions and Innovations in Model-Driven Software Engineering

Vicente García Díaz (University of Oviedo, Spain), Juan Manuel Cueva Lovelle (University of Oviedo, Spain), B. Cristina Pelayo García-Bustelo (University of Oviedo, Spain) and Oscar Sanjuán Martínez (University of Oviedo, Spain)

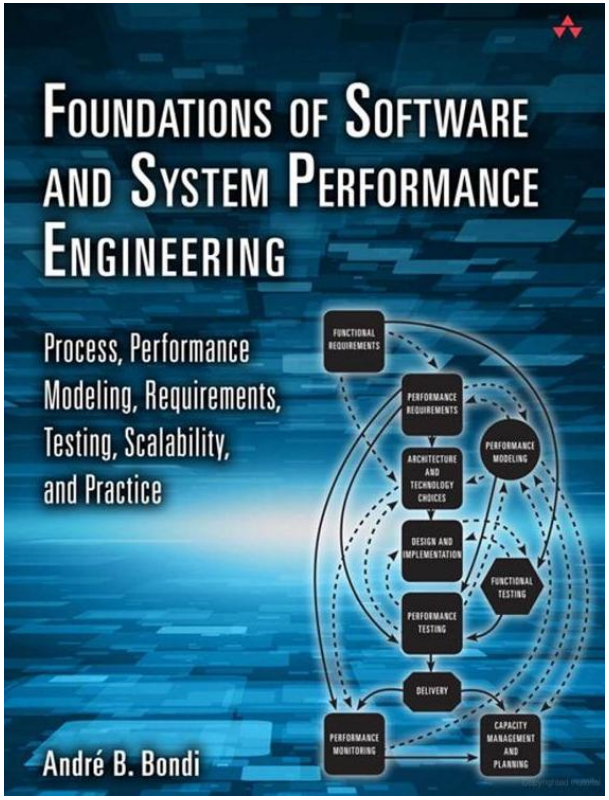
Release Date: June, 2013. Copyright © 2013. 388 pages.

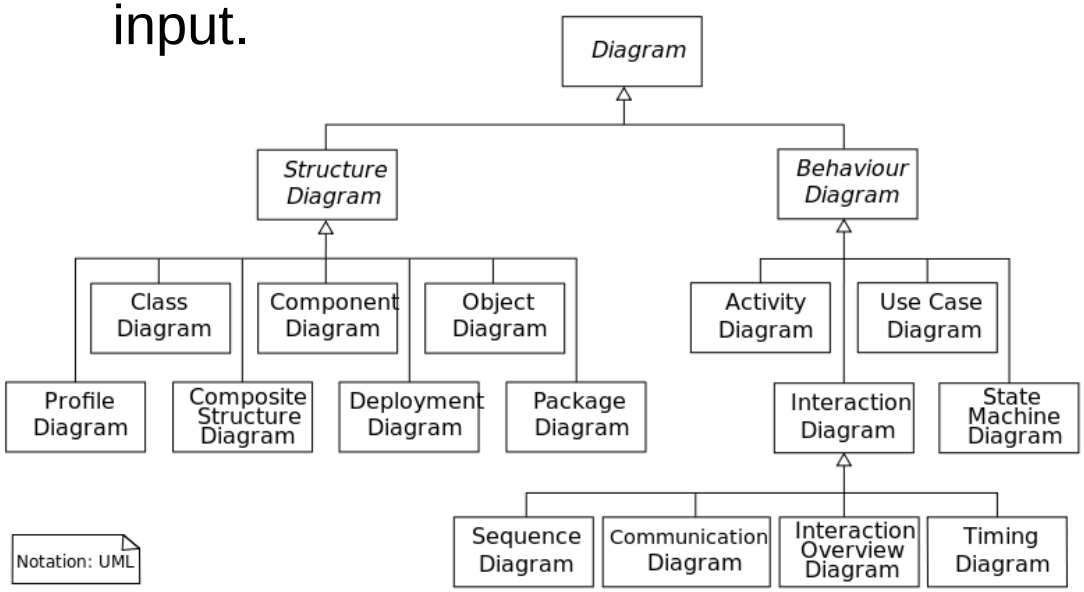
A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method
- Article selection
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Trends, bibliometrics and demographics
- Summary of findings, trends, and implications

An overview of UML-SPE

- **Software Performance Engineering (SPE)**
  - is a systematic and quantitative discipline to construct software systems that meet performance objectives.
- **UML-SPE**
  - is a family of SPE approaches that use UML models of a software system as input.

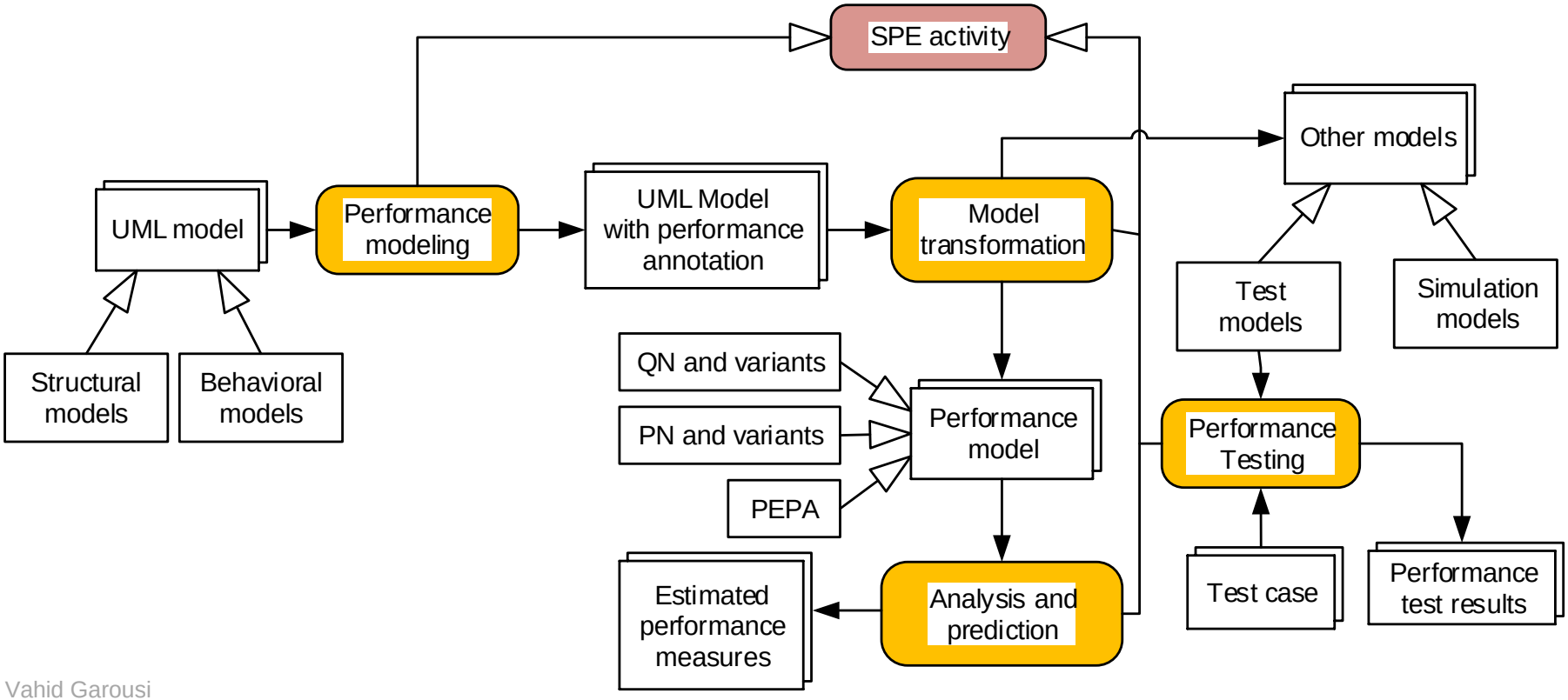




UNIFIED MODELING LANGUAGE™

An overview of UML-SPE

- **There are in general four types of SPE activities:**
  - (1) modeling (specifying) the performance properties, e.g., workload, inter-arrival time distribution on UML models
  - (2) model transformation, e.g., from UML to performance models, e.g., Queuing Networks (QN), Petri Nets (PN) and performance evaluation process algebra (PEPA)
  - (3) performance testing, e.g., load testing, and stress testing
  - (4) analysis and prediction of performance

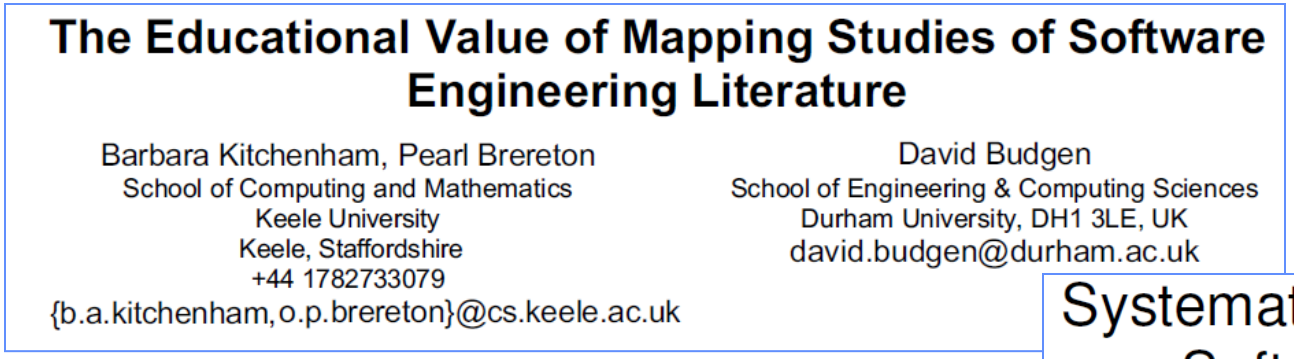


A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method for our SM
- Article selection for our SM
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Trends, bibliometrics and demographics
- Summary of findings, trends, and implications

Systematic mapping studies in software engineering

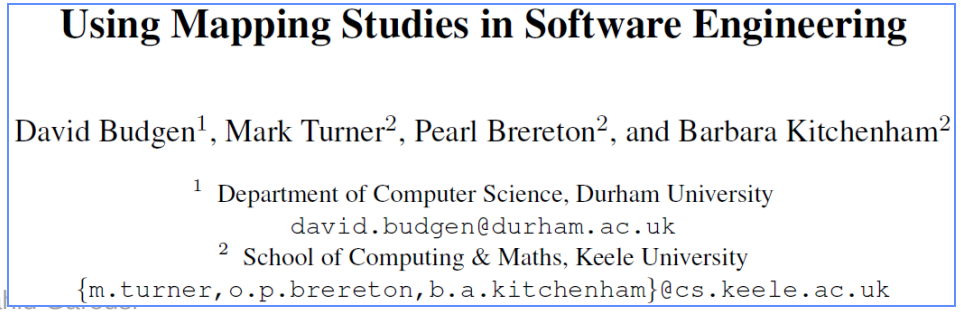
- **A software engineering systematic map is a defined method to build a classification scheme and structure a software engineering field of interest**
- **The analysis of results focuses on frequencies of publications for categories within the scheme**
- **Such summarized and categorized results provide many benefits to the broader community**
- **For example, they are valuable resources for new researchers (e.g., PhD students) aiming to conduct additional secondary studies**
- **Usually, a SM has less depth than a systematic literature review (SLR)**



The Educational Value of Mapping Studies of Software Engineering Literature

Barbara Kitchenham, Pearl Brereton
School of Computing and Mathematics
Keele University
Keele, Staffordshire
+44 1782733079
{b.a.kitchenham,o.p.brereton}@cs.keele.ac.uk

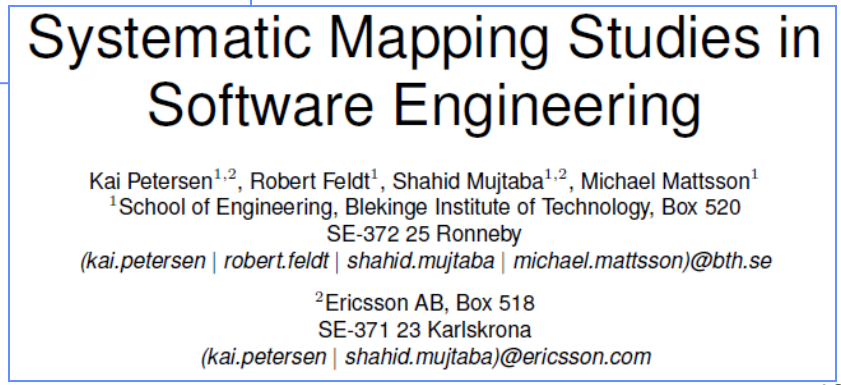
David Budgen
School of Engineering & Computing Sciences
Durham University, DH1 3LE, UK
david.budgen@durham.ac.uk



Using Mapping Studies in Software Engineering

David Budgen[1], Mark Turner[2], Pearl Brereton[2], and Barbara Kitchenham[2]

[1] Department of Computer Science, Durham University
david.budgen@durham.ac.uk
[2] School of Computing & Maths, Keele University
{m.turner,o.p.brereton,b.a.kitchenham}@cs.keele.ac.uk



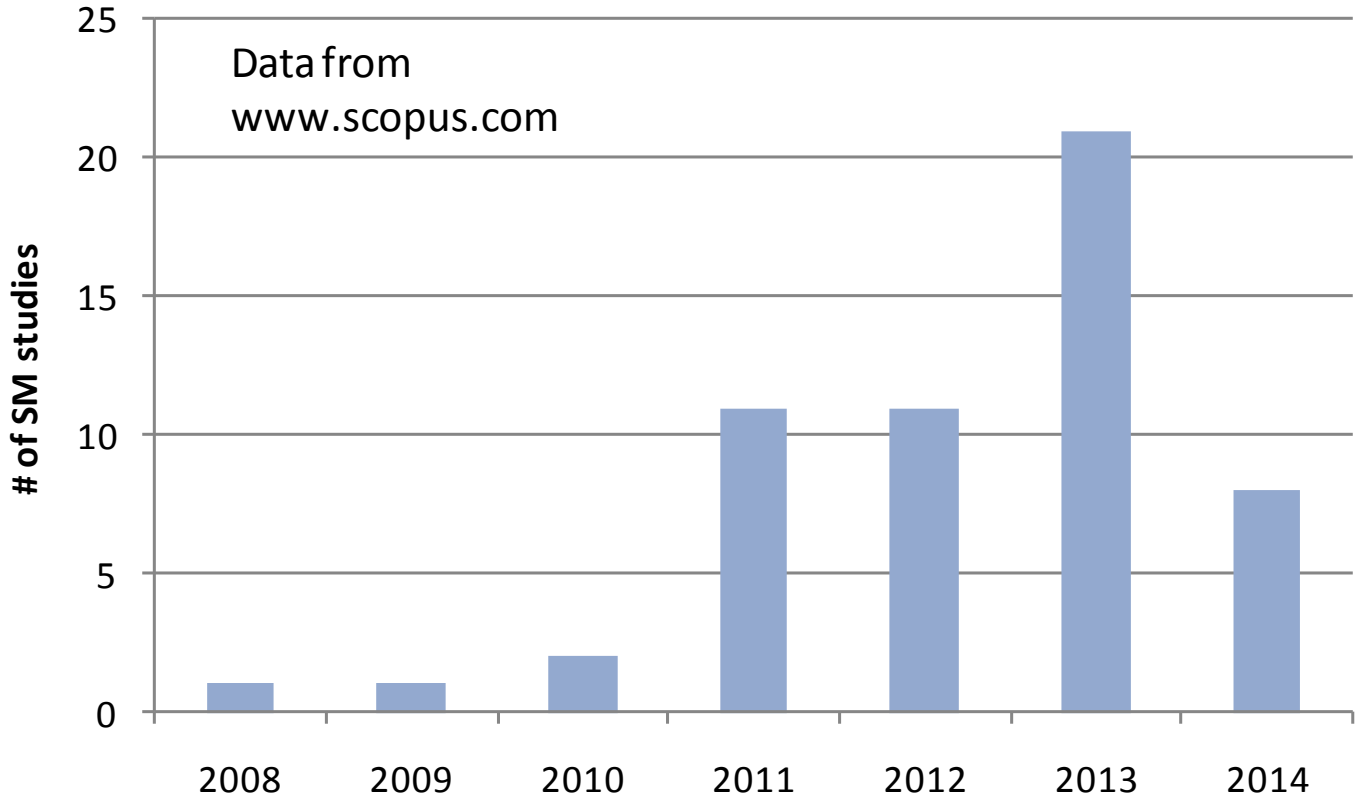
Systematic Mapping Studies in Software Engineering

Kai Petersen[1,2], Robert Feldt[1], Shahid Mujtaba[1,2], Michael Mattsson[1]
[1]School of Engineering, Blekinge Institute of Technology, Box 520
SE-372 25 Ronneby
(kai.petersen | robert.feldt | shahid.mujtaba | michael.mattsson)@bth.se

[2]Ericsson AB, Box 518
SE-371 23 Karlskrona
(kai.petersen | shahid.mujtaba)@ericsson.com

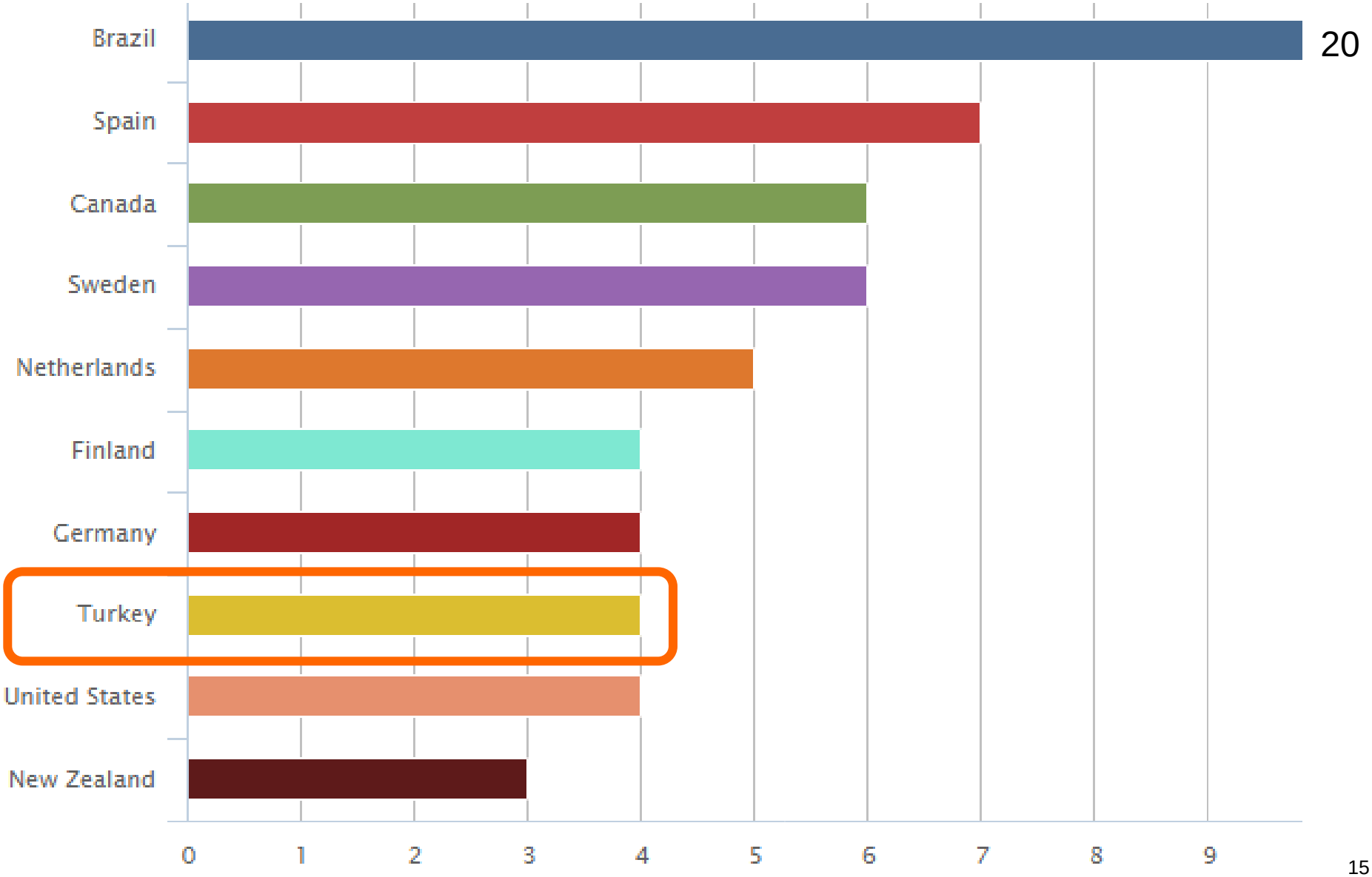
Systematic mapping studies in software engineering

- **More and more systematic mapping studies are appearing…**
- **There are about 55 SE systematic mapping studies in Scopus as of Oct. 2014**
- **Query:**
  - Title (systematic mapping) AND SourceTitle (software)



Systematic mapping studies in software engineering

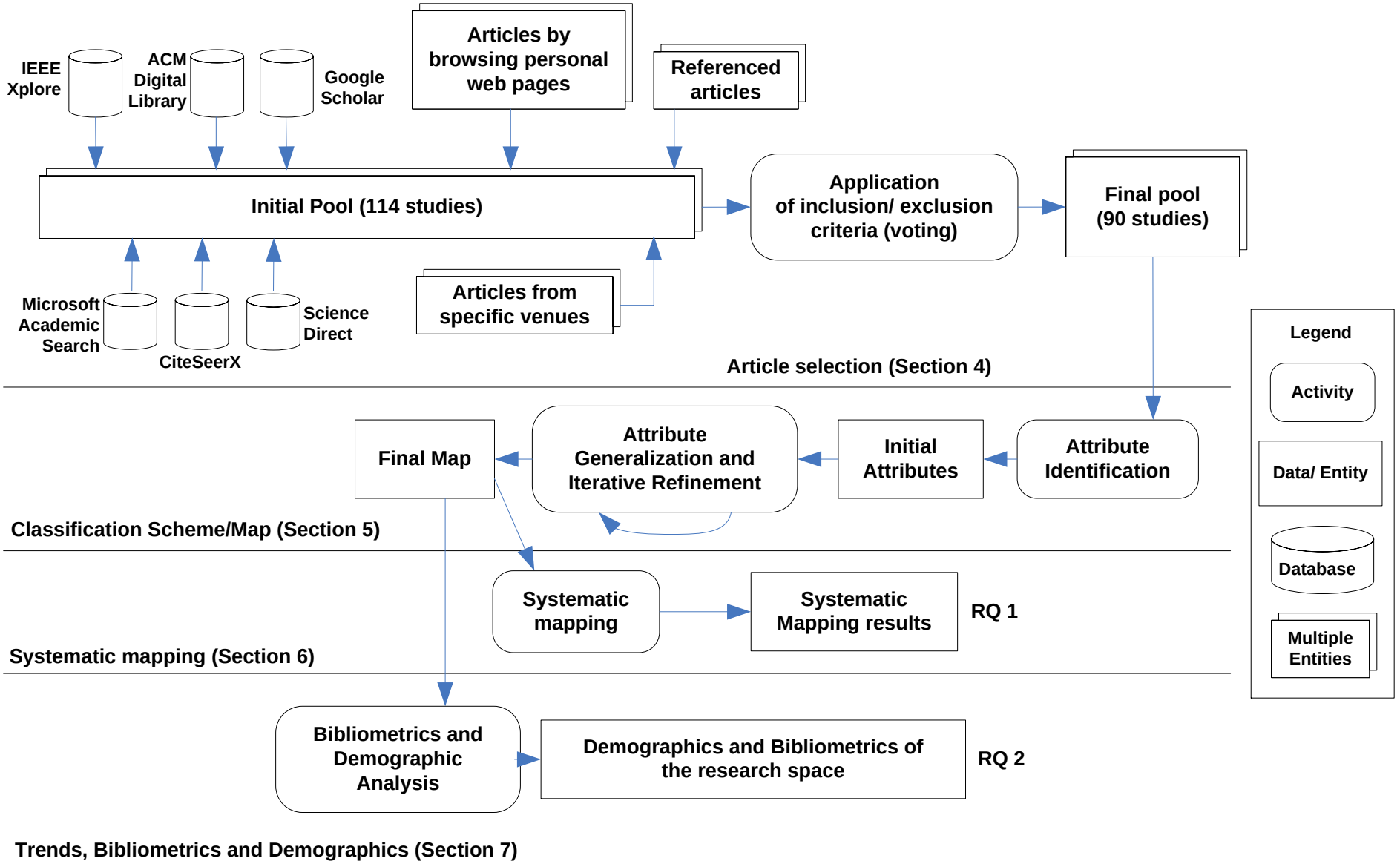
- **Active countries…**



A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method for our SM
- Article selection for our SM
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Trends, bibliometrics and demographics
- Summary of findings, trends, and implications

Research method for our SM



Research method for our SM

- **Goal:**
  - The goal of was to systematically map (classify) the state-of-the-art in the area of UML-SPE,
  - to identify opportunities for future research,
  - to explore the bibliometrics trends in this area,
  - and to find out the recent trends and directions in this field,
  - from the point of view researchers and practitioners in this area.
- **Research Questions:**
  - RQ 1 - Systematic mapping: What is the research space of UML-SPE? This question aims at conducting a systematic mapping (classification) in the area.
  - RQ 2 – What are the bibliometrics and demographics in this area?

Each RQ was divided into sub-questions

- **RQ 1 - What is the research space of UML-SPE? This question aims at conducting a systematic mapping (classification) in the area.**
  - **RQ 1.1-Mapping of studies by contribution facet:** How many studies present UML-SPE methods, techniques, tools, models, metrics, or processes? Petersen et al. [95] proposed the above types of contributions to enable systematic mapping of studies in software engineering.
  - **RQ 1.2- Mapping of studies by research facet:** What type of research methods are used in the studies in this area? Some studies only propose solutions without extensive validations, while some other studies present in-depth evaluation of their approach. Petersen et al. [95] has also proposed guidelines to classify the research approach of papers, which we will use to answer this RQ.
  - **RQ 1.3- Types of SPE approach:** What types of SPE activities have been presented in the literature (e.g., performance modeling, model transformation, and testing), and which types are more popular than others?
  - **…**
  - **RQ 1.10 – Attributes of the software systems under analysis:** What are the attributes of the software systems under analysis in the studies? What ratios of studies have used open-source, commercial, or academic experimental systems for evaluation?
  - **RQ 1.11 – Tools presented in papers:** How many SPE tools have been proposed in the studies? Are they available for download and/or purchase?

Each RQ was divided into sub-questions

- **RQ 2 – What are the bibliometrics and demographics in this area?**
  - **RQ 2.1 - Publication count by year:** What is the annual publication count in this area?
  - **RQ 2.2 - Publication count by venue type:** What is the annual publication count in different types of venues (conference, journals, etc.)?
  - **RQ 2.3- Citation count by year:** What is the citation count for studies from different years? Do older papers necessarily receive more citations?
  - **RQ 2.4–Top-cited studies:** What are the top-cited studies in this area?
  - **RQ 2.5- Top venues:** Which venues have published most of the studies in this area?
  - **RQ 2.6– Citation count by venue type:** What is the average citation count for different publication venue types? Do journal papers in this area necessarily receive more citations than workshop and conference papers?
  - **RQ 2.7- Top authors:** Which authors have been most active in terms of number of papers?
  - **RQ 2.8- Author affiliation** What ratios of the authors are from academia or industry? How many papers have been jointly authored by people from academia and industry? This RQ will show the extent of academia- industry collaborations in this field.
  - **RQ 2.9- Top countries:** Which countries have been more active in terms of number of papers?

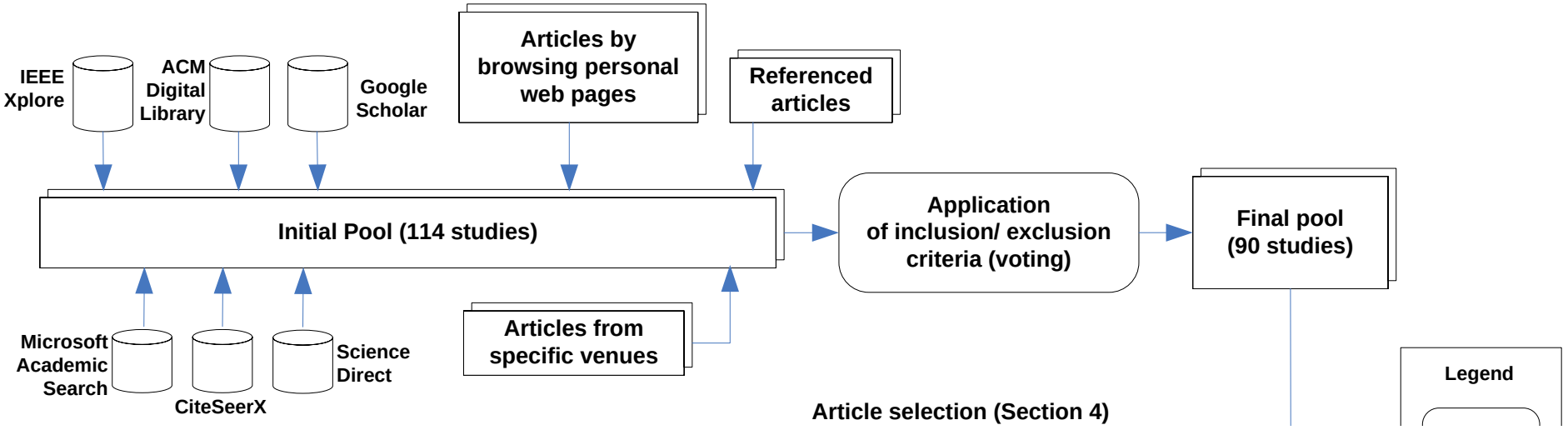
11+9=20 RQs in total

A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method for our SM
- Article selection for our SM
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Trends, bibliometrics and demographics
- Summary of findings, trends, and implications

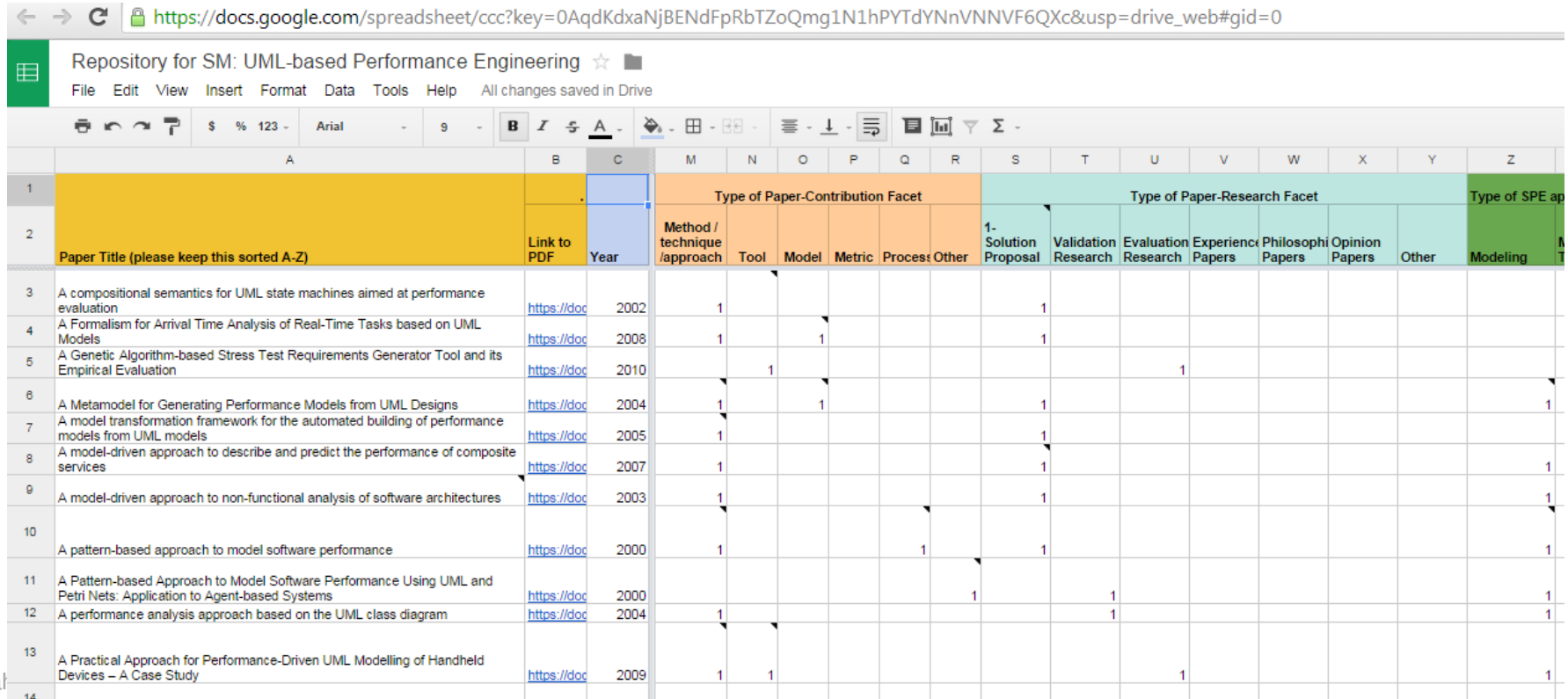
Article selection for our SM

- **Search keywords:**
  - UML performance engineering
  - UML performance modeling
  - UML performance testing
  - UML load testing
  - UML performance prediction
  - UML stress testing



Article selection for our SM

- **Final pool of articles:**
  - After the initial search and the follow-up analysis for exclusion of unrelated and inclusion of additional studies, the pool of selected studies was finalized with 90 studies.
- **Online repository:**
  - http://goo.gl/V8ltgV



https://docs.google.com/spreadsheet/ccc?key=0AqdKdxaNjBENdFpRbTZoQmg1N1hPYTdYNnVNNVF6QXc&usp=drive_web#gid=0

Repository for SM: UML-based Performance Engineering

File Edit View Insert Format Data Tools Help All changes saved in Drive

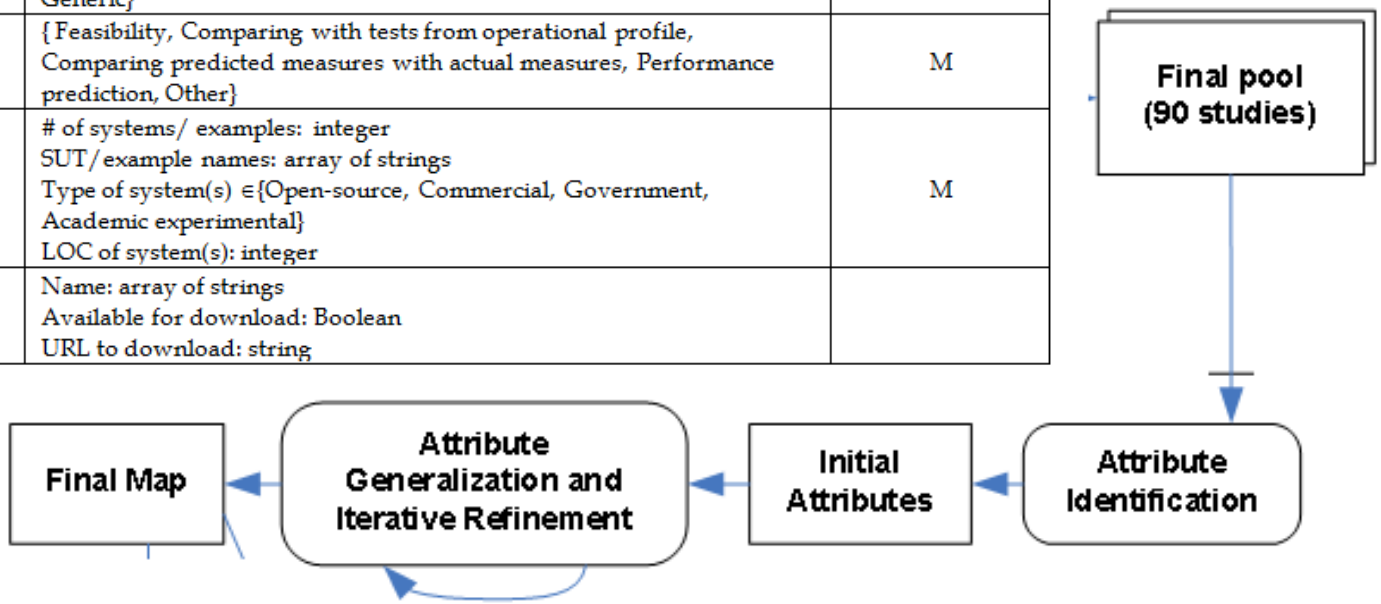
	Paper Title (please keep this sorted A-Z)	Link to PDF	Year	Type of Paper-Contribution Facet: Method / technique /approach	Tool	Model	Metric	Process	Other	Type of Paper-Research Facet: 1- Solution Proposal	Validation Research	Evaluation Research	Experience Papers	Philosophi Papers	Opinion Papers	Other	Type of SPE ap: Modeling
3	A compositional semantics for UML state machines aimed at performance evaluation	https://doc	2002	1						1							
4	A Formalism for Arrival Time Analysis of Real-Time Tasks based on UML Models	https://doc	2008	1		1				1							
5	A Genetic Algorithm-based Stress Test Requirements Generator Tool and its Empirical Evaluation	https://doc	2010		1							1					
6	A Metamodel for Generating Performance Models from UML Designs	https://doc	2004	1		1				1							1
7	A model transformation framework for the automated building of performance models from UML models	https://doc	2005	1						1							
8	A model-driven approach to describe and predict the performance of composite services	https://doc	2007	1						1							1
9	A model-driven approach to non-functional analysis of software architectures	https://doc	2003	1						1							1
10	A pattern-based approach to model software performance	https://doc	2000	1				1		1							1
11	A Pattern-based Approach to Model Software Performance Using UML and Petri Nets: Application to Agent-based Systems	https://doc	2000						1		1						1
12	A performance analysis approach based on the UML class diagram	https://doc	2004	1							1						1
13	A Practical Approach for Performance-Driven UML Modelling of Handheld Devices – A Case Study	https://doc	2009	1	1							1					1

A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method for our SM
- Article selection for our SM
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Trends, bibliometrics and demographics
- Summary of findings, trends, and implications

Development of the systematic map (classification scheme)

RQ	Attribute/Aspect	Categories	(M)ultiple/ (S)ingle
1.1	Contribution type	{Method (technique), Tool, Metric, Process, Other }	M
1.2	Research type	{Solution Proposal, Validation Research, Evaluation Research, Experience Papers, Philosophical Papers, Opinion Papers, Other}	S
1.3	Type of SPE approach	{Modeling, Model Transformation, Testing, Analysis and prediction, Other}	M
1.4	Type of performance metrics used and evaluated	{Execution (response) Time, Utilization, Throughput, Other}	M
1.5	Input UML diagrams	{Activity diagram, Class diagram, Collaboration Diagram, Deployment Diagram, Sequence Diagram, States Machine, Use-case diagram, Other}	M
1.6	Outputs	{Performance model, test cases, other} Performance model ∈{queuing network (QN) and variants, Petri net (PN) and variants, Performance Evaluation Process Algebra, other}	M
1.7	Annotation languages	{ UML Profile for Schedulability, Performance and Time (SPT), UML Profile for Modeling and Analysis of Real-Time and Embedded Systems (MARTE), other/custom}	M
1.8	Application domains	{ Real-time systems, Embedded Systems, Mobile/handheld, Other, Generic}	M
1.9	Methods of evaluation	{ Feasibility, Comparing with tests from operational profile, Comparing predicted measures with actual measures, Performance prediction, Other}	M
1.10	Attributes of the software systems under analysis	# of systems/ examples: integer SUT/example names: array of strings Type of system(s) ∈{Open-source, Commercial, Government, Academic experimental} LOC of system(s): integer	M
1.11	Attributes of the tool(s) presented in the paper (if any)	Name: array of strings Available for download: Boolean URL to download: string	

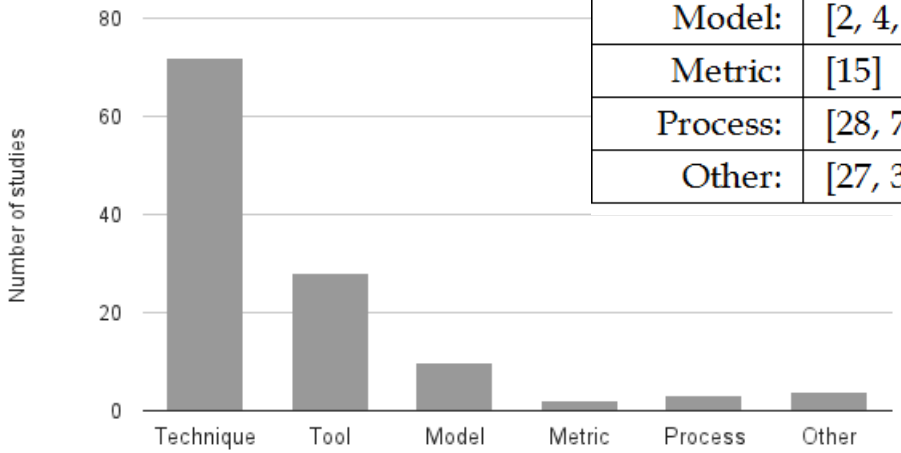


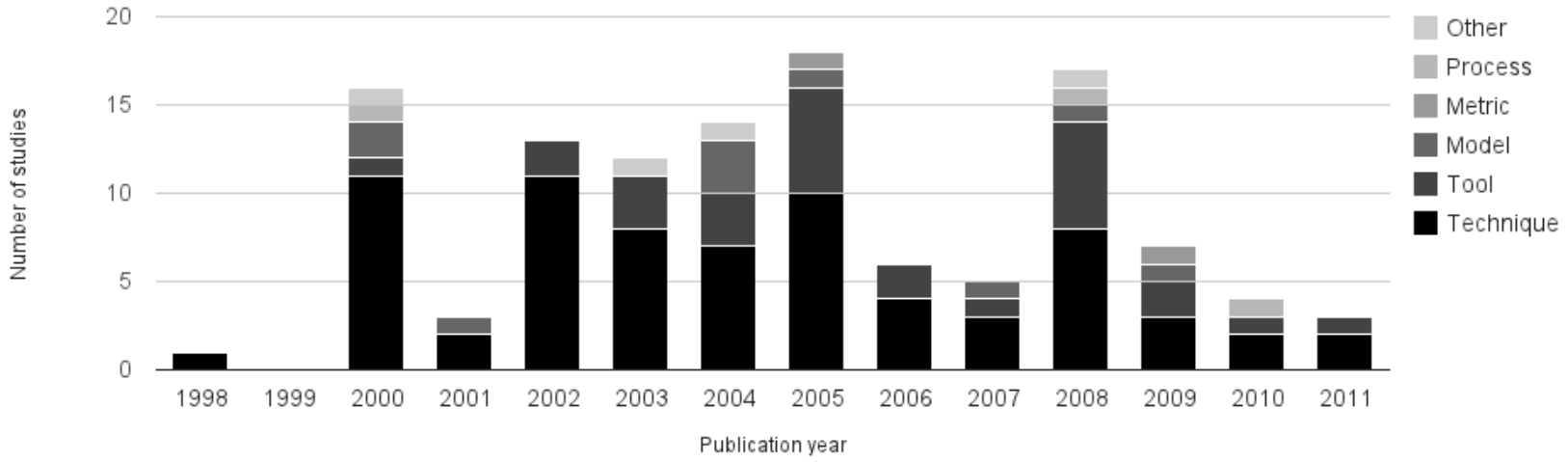
A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method
- Article selection
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Bibliometrics and demographics
- Summary of findings, trends, and implications

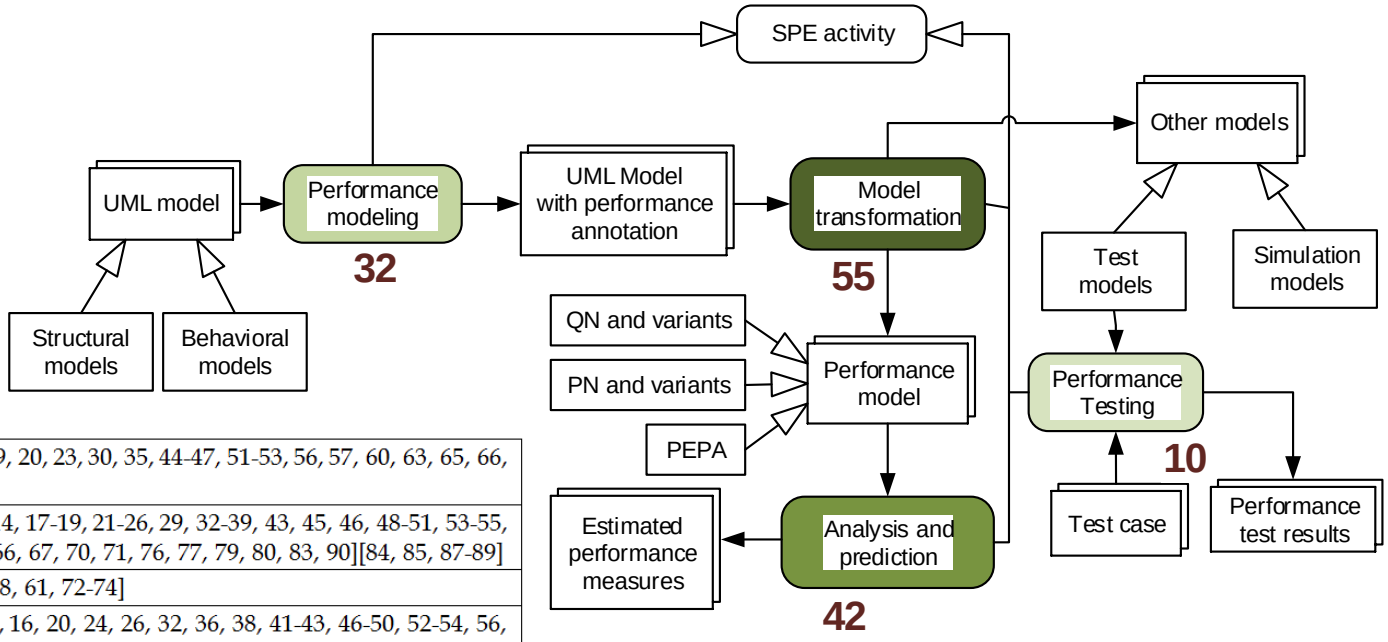
Mapping of Studies by Contribution Facet (RQ 1.1)

Technique:	[1, 2, 5-11, 15, 18-20, 22-26, 28, 29, 31-49, 51, 53-59][61-63, 65-77, 80, 82-87, 89]
Tool:	[3, 10, 13, 14, 16, 19, 21, 22, 27, 31-33, 49, 50, 59-61, 64, 66-69, 73, 78, 84-86, 88]
Model:	[2, 4, 15, 17, 35, 51, 58, 76, 79, 81]
Metric:	[15]
Process:	[28, 71, 90]
Other:	[27, 30, 52, 90]

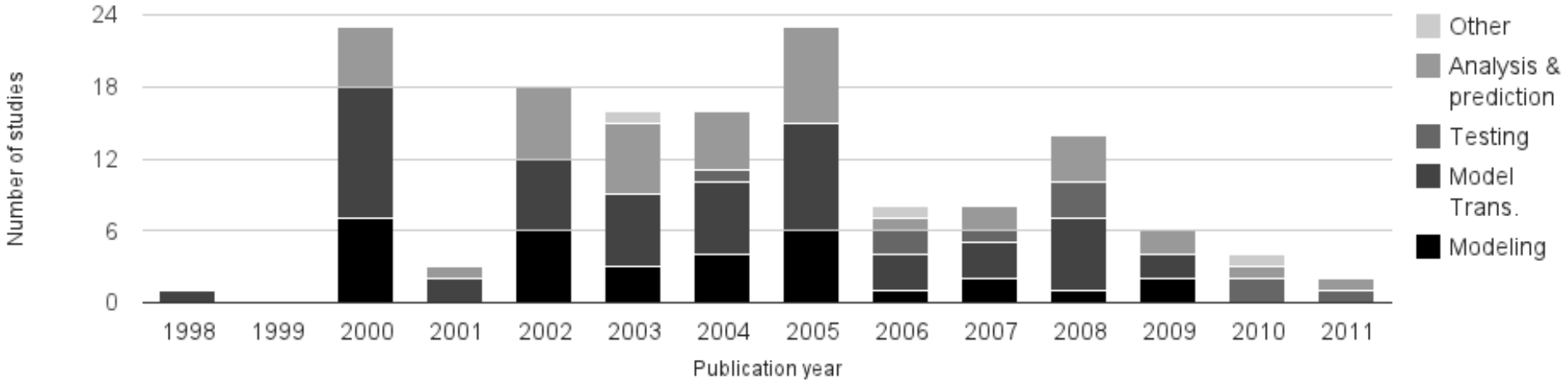




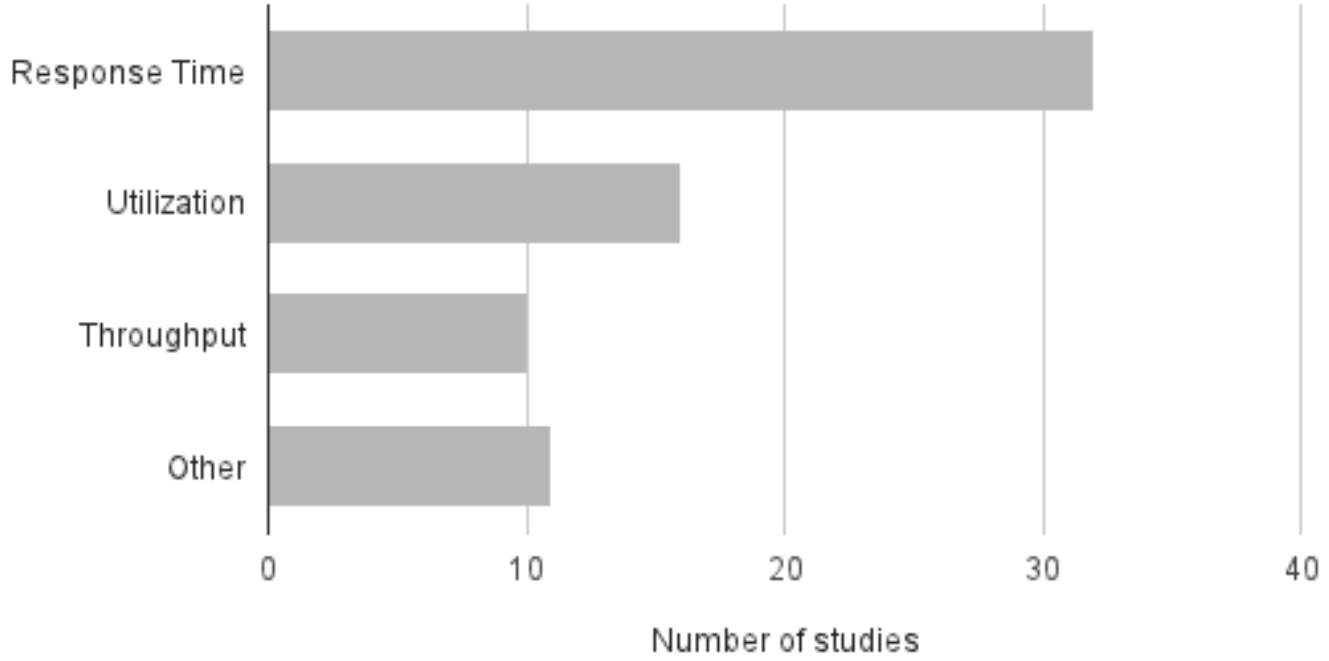
Type of SPE activities (RQ 1.3)



Modeling:	[4-6, 8-10, 16, 17, 19, 20, 23, 30, 35, 44-47, 51-53, 56, 57, 60, 63, 65, 66, 68, 75, 81, 86, 90]
Model Transformation:	[1, 4, 7, 10, 11, 13, 14, 17-19, 21-26, 29, 32-39, 43, 45, 46, 48-51, 53-55, 57, 59, 61, 63, 64, 66, 67, 70, 71, 76, 77, 79, 80, 83, 90][84, 85, 87-89]
Testing:	[3, 11, 27, 28, 31, 58, 61, 72-74]
Analysis and prediction:	[2, 5, 6, 8-10, 12, 14, 16, 20, 24, 26, 32, 36, 38, 41-43, 46-50, 52-54, 56, 57, 60, 62, 65, 66, 68, 69, 71, 76-78, 82, 84-86]
Other:	[28, 40, 46]



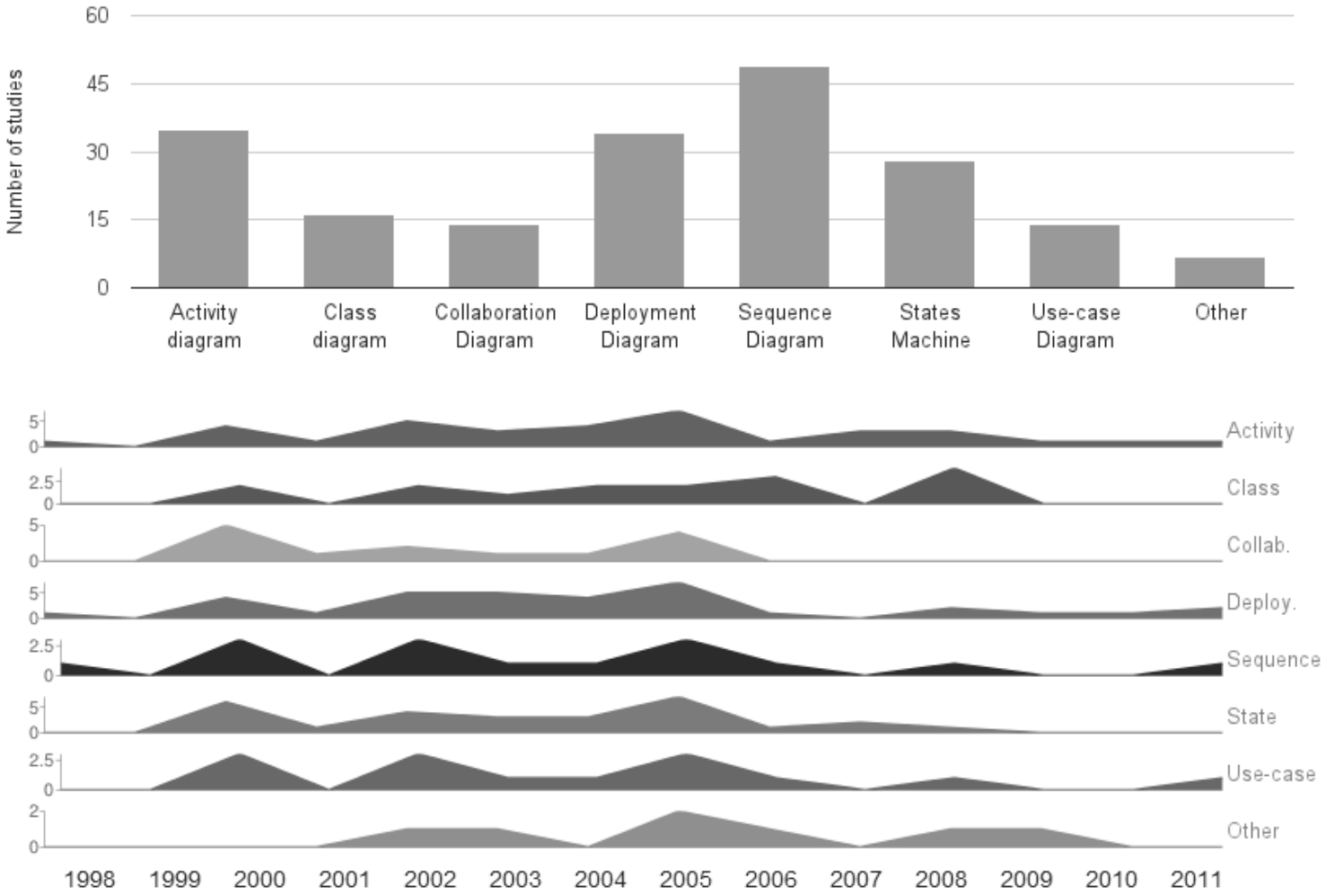
Type of Performance Metrics Used and Evaluated (RQ 1.4)



- **Other types of performance metrics, e.g., queue length [52, 86], sojourn time [54] (amount of time it takes for an object to leave the system), probability of missing a deadline [46], and network transfer rate [50]**

Response time:	[2, 3, 5, 10, 11, 26-28, 31, 32, 38, 41, 42, 46, 48-54, 56-58, 60, 62, 65, 68, 72-74, 84, 90]
Utilization:	[12, 15, 26, 29, 40, 46-49, 52, 56, 63, 65, 66, 69, 85]
Throughput:	[12, 23, 26, 51, 54, 63, 66, 69, 76, 77]
Other:	[13, 15, 16, 21, 36, 46, 50, 52, 54, 65, 86]

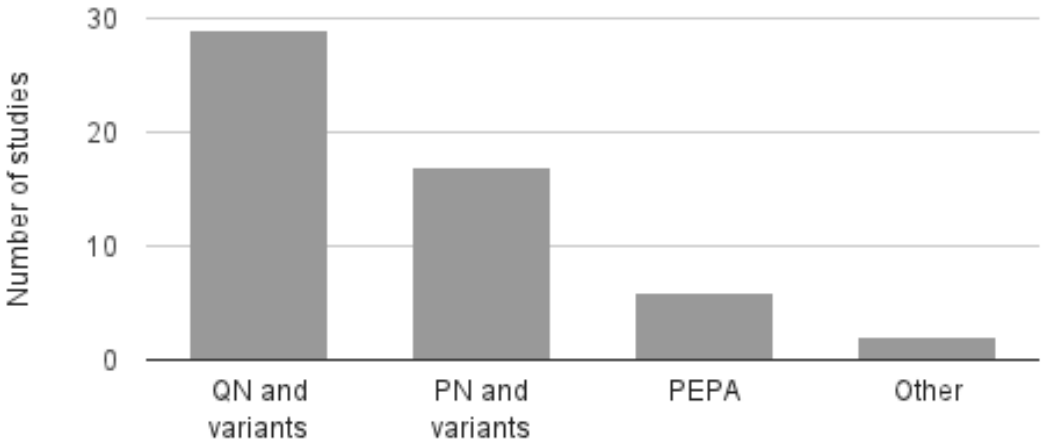
Type of input UML diagrams (RQ 1.5) used by techniques



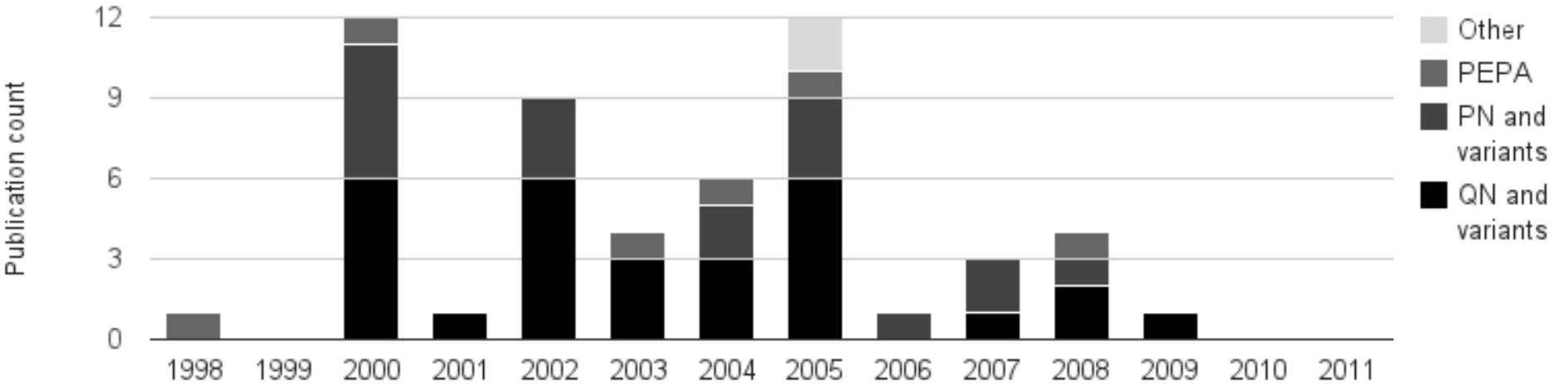
- 6 studies used other types of UML diagrams, e.g., component diagrams [32, 48, 68], and composite-structure diagrams [10]; or extended UML diagrams such as: use-case maps [67].

Output Models and Artifacts (RQ 1.6)

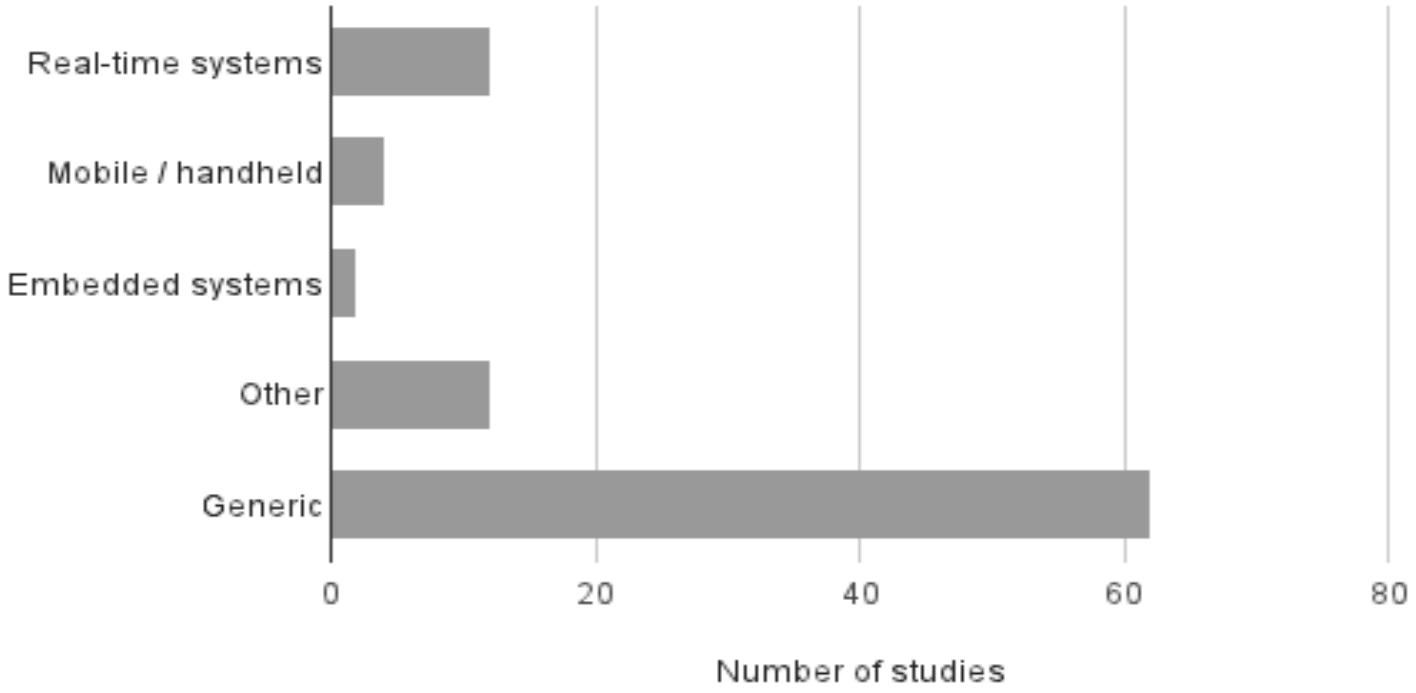
- **The top three performance models are:**
  - (1) Queuing Network (QN) and variants
  - (2) Petri Net (PN) and variants
  - (3) Performance Evaluation Process Algebra (PEPA)



- **Various variants of QN included:**
  - Layered QN (LQN) (e.g., [7])
  - Multi-class QN (MCQN) [10, 55]
  - Extended QN model (EQNM) [24, 62, 63]
  - Augmented QN (AQN) [76]



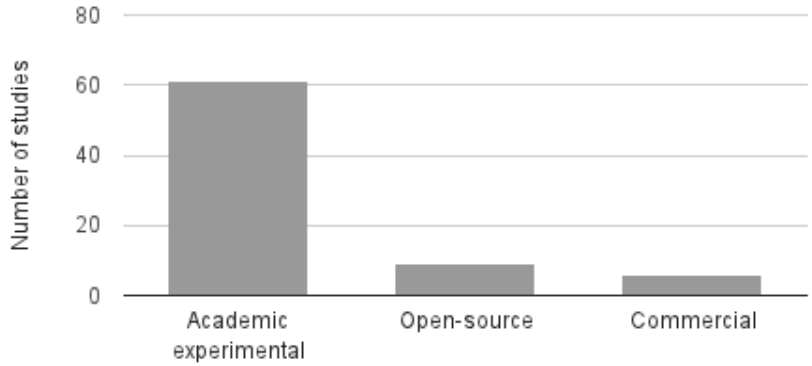
Application Domains (RQ 1.8)

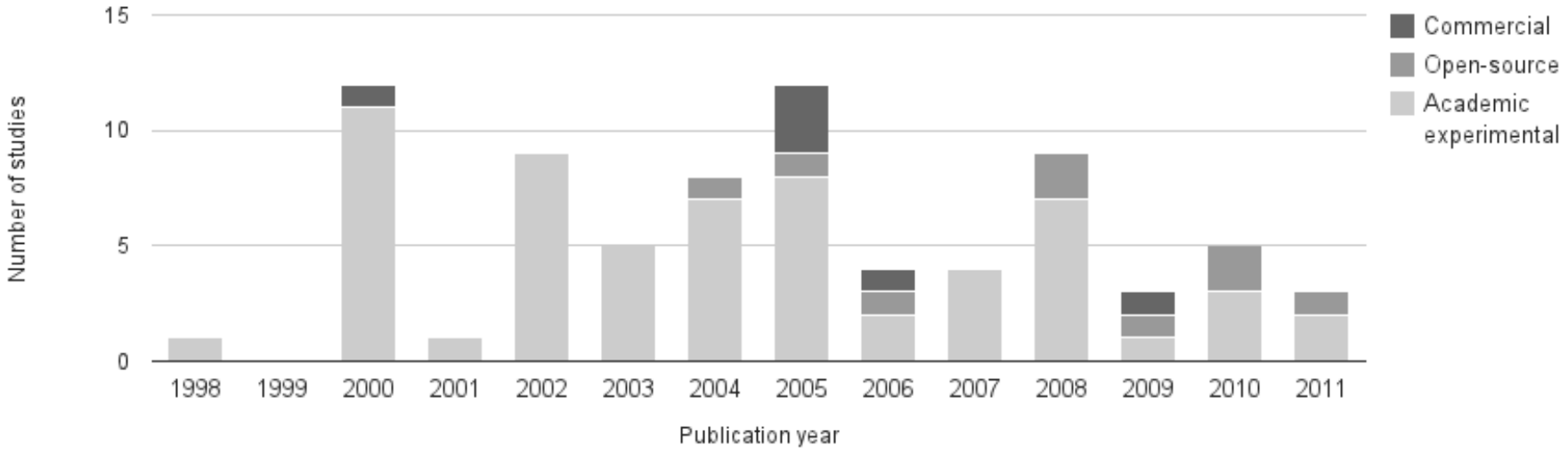


- **The “other” domains:**
  - parallel and distributed applications [44, 60, 75-77]
  - agent-based systems [53]
  - enterprise information systems [42]

Type/Scale of the Software Systems Under Analysis (RQ 1.10)

- **The following commercial software systems or protocols have been used in the studies:**
  - Digital Broadcasting Video (DVB) protocol (used in [10])
  - Siemens medical solutions (used in [11])
  - A hierarchical cellular network [16]
  - Alternating bit protocol [23]
  - NASA's Earth Observing System (EOS) [41]
  - …



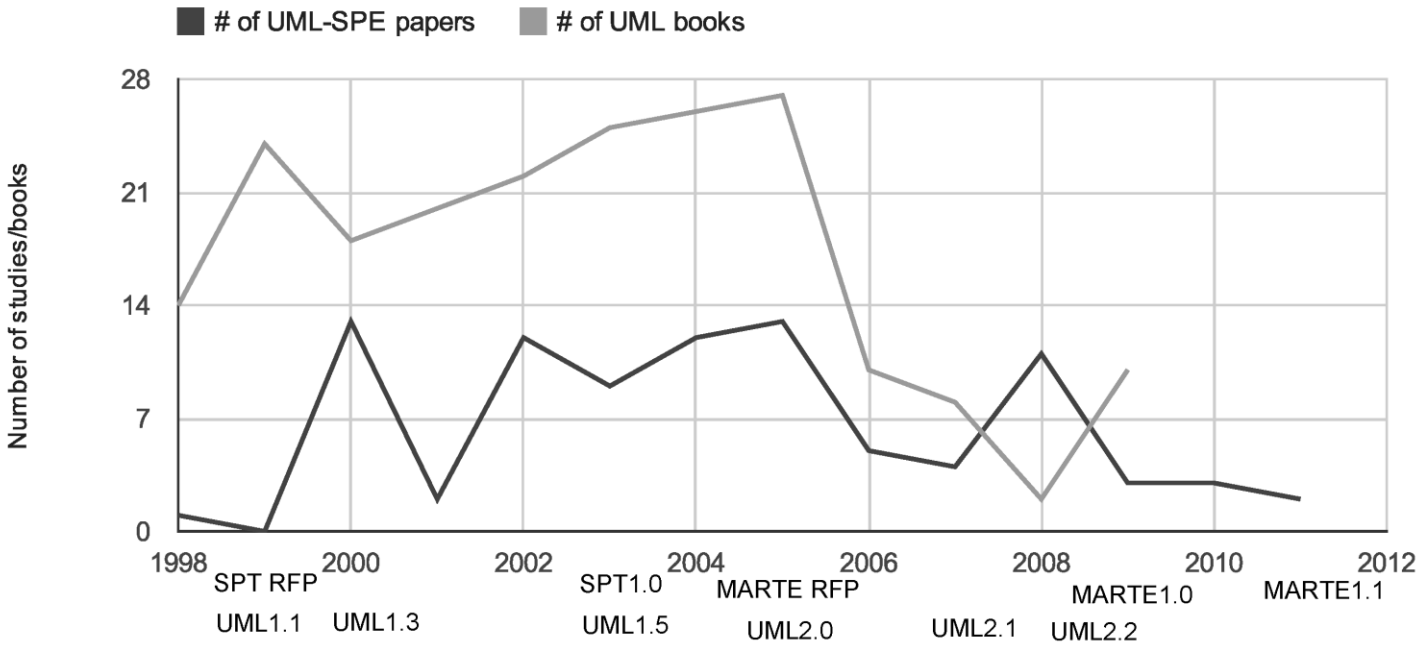


A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method
- Article selection
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Bibliometrics and demographics
- Summary of findings, trends, and implications

Publication Count by Year (RQ 2.1)

- **The annual trend of number of papers has had a decline in recent years.**
- **This is raising the question that whether there is very little left in the field to be "solved".**



- **Also, compared with the trend of UML books from another SM study**

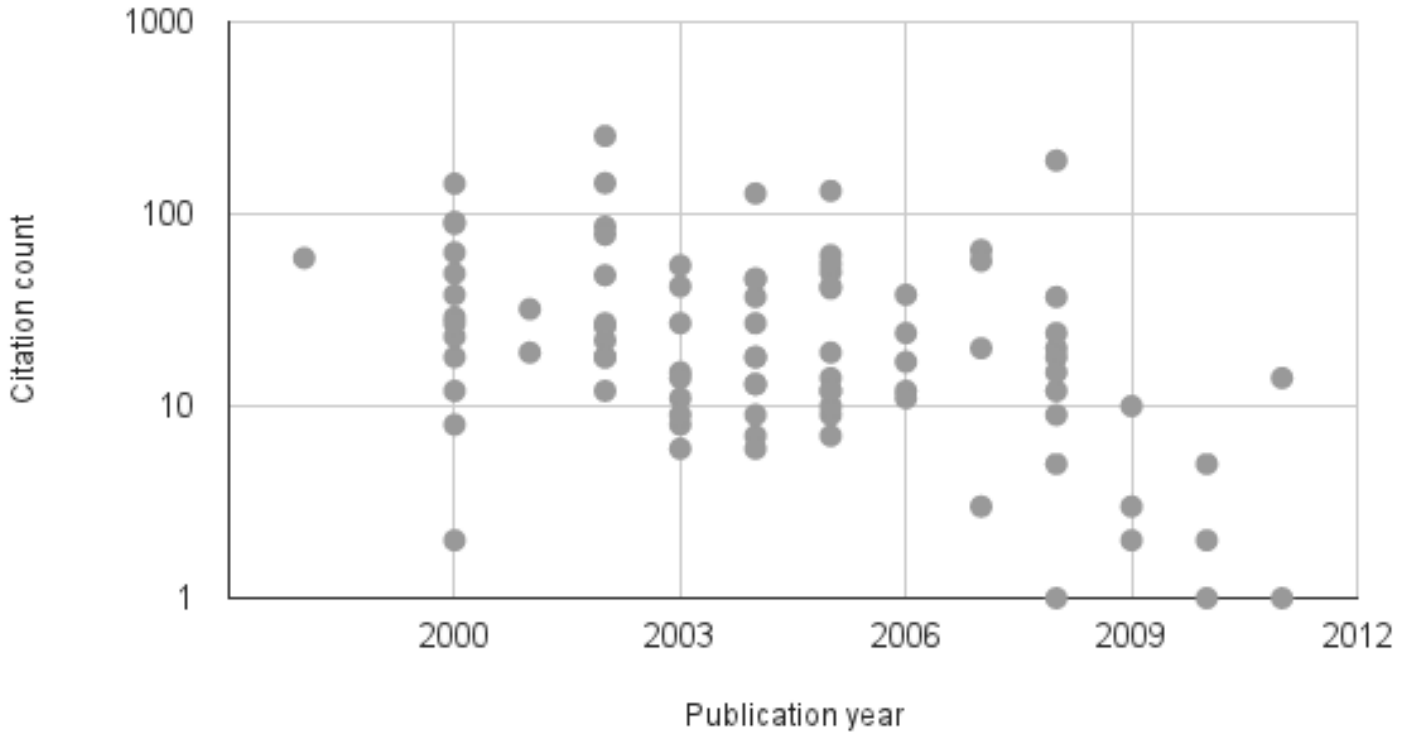
Softw Syst Model
DOI 10.1007/s10270-011-0189-9

OVERVIEW PAPER

Classification and trend analysis of UML books (1997–2009)

Vahid Garousi

Citation Count versus Publication Year (RQ 2.3)

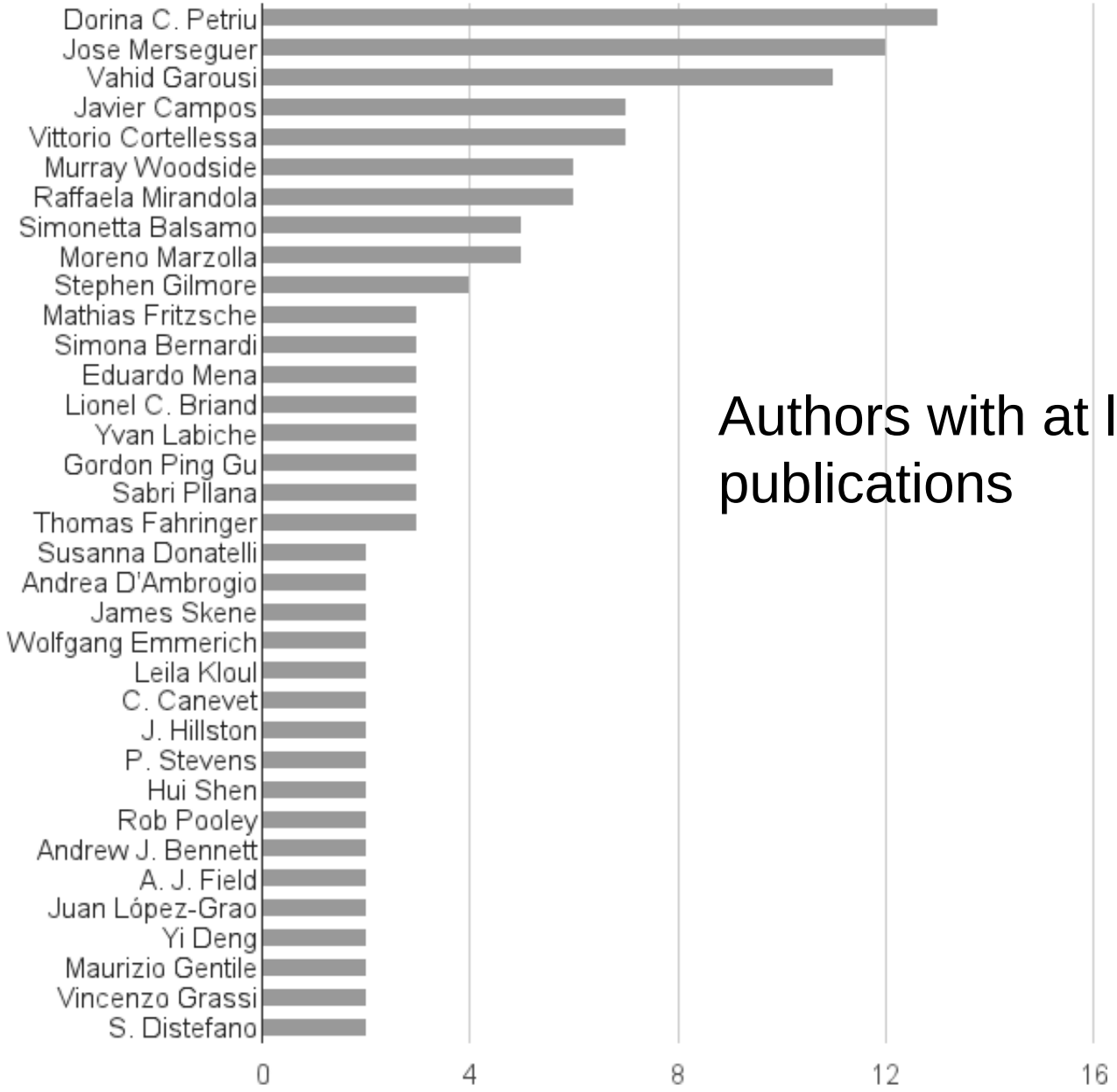


- **To analytically assess the hypothesis that whether older papers necessarily receive more citations, we calculated the Pearson correlation coefficient.**
- **The correlation coefficient=-0.25**
- **which denote that there is a weak support for the above hypothesis**

Top-Cited Studies (RQ 2.4)

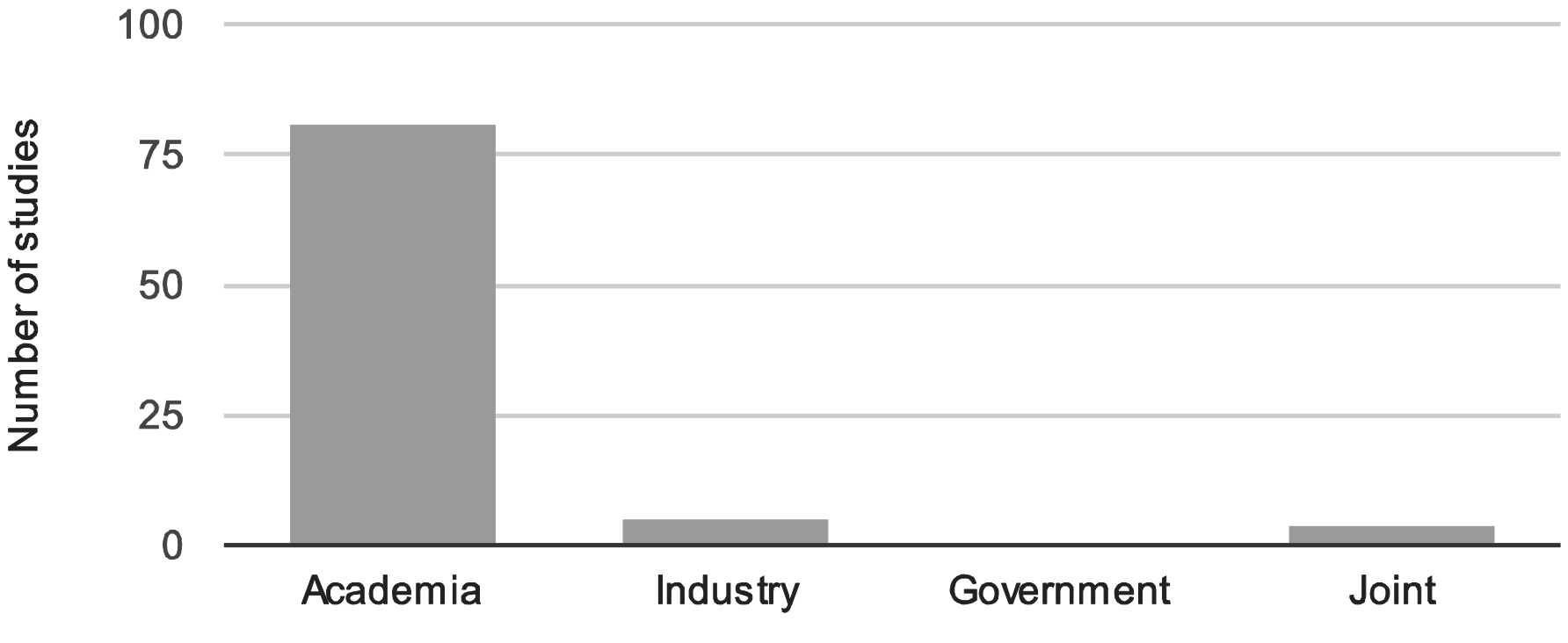
Title	Number of citations	Publication year	Type of SPE approach
From UML sequence diagrams and state-charts to analyzable Petri net	255	2002	Model Transformation
The Palladio component model for model-driven performance prediction	190	2008	Model Transformation
Applying the UML Performance Profile: Graph Grammar-Based Derivation of LQN Models from UML Specifications	145	2002	Modeling, Analysis and prediction

Active Authors (RQ 2.7)

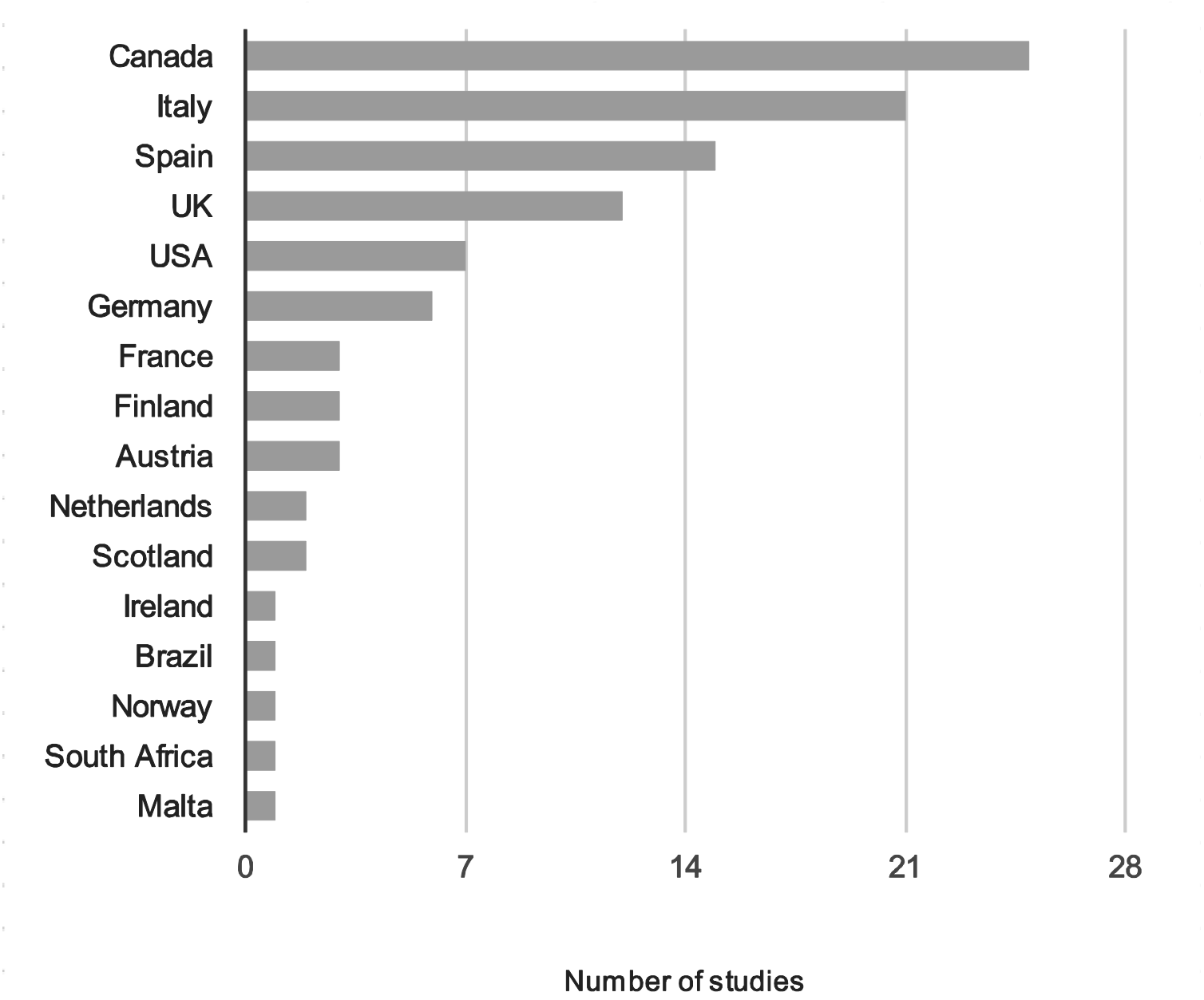


Authors with at least two publications

Authors Affiliation (RQ 2.8)



Top Countries (RQ 2.9)



A systematic mapping on UML-SPE

- An overview of UML-SPE
- An overview of systematic mapping studies in software engineering
- Research method
- Article selection
- Development of the systematic map (classification scheme)
- Results of systematic mapping
- Bibliometrics and demographics
- Summary of findings, trends, and implications

Summary of findings, trends, and implications

- **RQ 1 (mapping of studies by contribution facet):**
  - In terms of contribution facet, most of the primary studies (81%) proposed new techniques or improved an existing one.
  - Only 1 and 3 studies presented new metrics and new processes, respectively.
  - This denotes the need for more work in these areas.
- **RQ 1.3 (types of SPE approach):**
  - The ranking of proposed SPE activities in order were: (1) model transformation (61% of studies), (2) analysis and prediction (47%), (3) modeling (36%), (4) testing (10%), and (5) other (3%), e.g., performance tuning.
  - While UML-based performance testing is an important and promising field of study [28, 72, 73], this particular area has not received the deserved attention and focus in the community, and thus needs more work by the research community and practitioners.

Summary of findings, trends, and implications

- **RQ 1.7 (model annotation languages):**
  - 43 studies have used the UML SPT profile [116, 117] as their annotation languages
  - while 5 studies have used the UML MARTE profile [118]
  - The annual usage trend analysis reveals that usage of SPT continues through the years.
- **RQ 1.10 (type/scale of the software systems under analysis):**
  - Our results showed that prototype or experimental systems developed in the academia are the majority (used in 62 studies).
  - 9 and 8 studies used open-source or commercial software for evaluation of their methods.
  - We feel this opens up a fertile opportunity for future work to evaluate the UML-SPE techniques proposed on large-scale commercial software systems.

Summary of findings, trends, and implications

- **RQ 1.11 (tools presented in papers):**
  - Tool support is an important issue in all sub-fields of the software engineering.
  - A “healthy” ratio of the primary studies (31%) presented 28 (new) tools.
  - Only 20 of those 28 tools were available for download (either free or commercial license).
  - Certainly, the authors encourage more efforts on developing additional industry-scale UML-SPE tools and also technology transfer of those tools to the industry.

Outline

- **Background of the speaker and his research expertise**
- **A systematic mapping of UML-Driven Software Performance Engineering (UML-SPE)**
- **Review of one UML-SPE technique**

Review of several selected techniques

- **The speaker has had 11 papers in this area.**

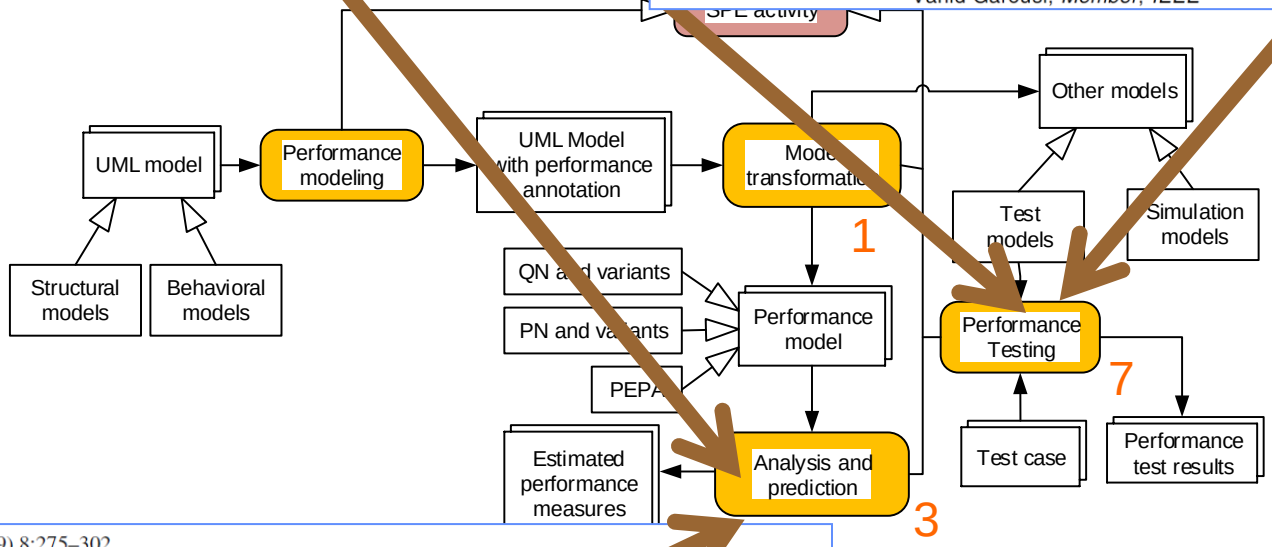
Experience and challenges with UML-driven performance engineering of a Distributed Real-Time System

Vahid Garousi

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 36, NO. 6, NOVEMBER/DECEMBER 2010

A Genetic Algorithm-Based Stress Test Requirements Generator Tool and Its Empirical Evaluation

Vahid Garousi, *Member, IEEE*



Softw Syst Model (2009) 8:275–302
DOI 10.1007/s10270-008-0099-7

REGULAR PAPER

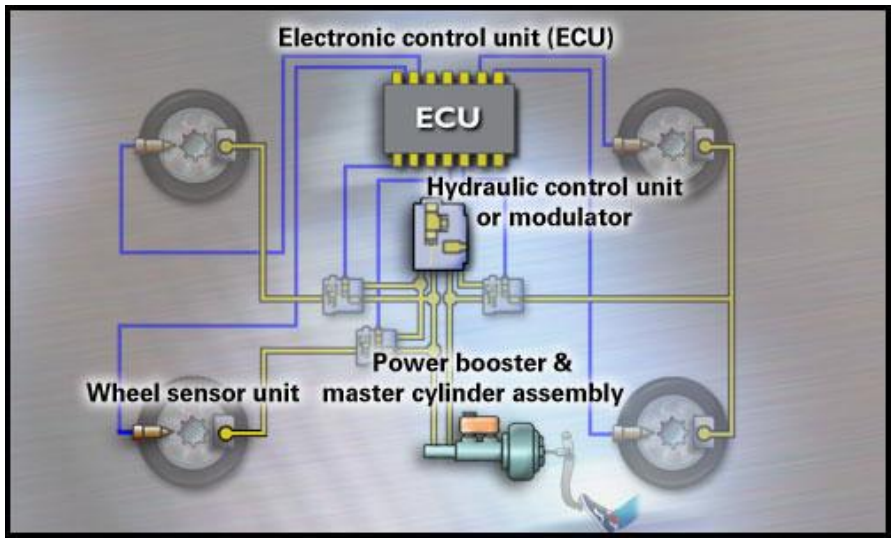
A UML-based quantitative framework for early prediction of resource usage and load in distributed real-time systems

Vahid Garousi · Lionel C. Briand · Yvan Labiche

Experience and challenges with UML-driven performance engineering of a Distributed Real-Time System

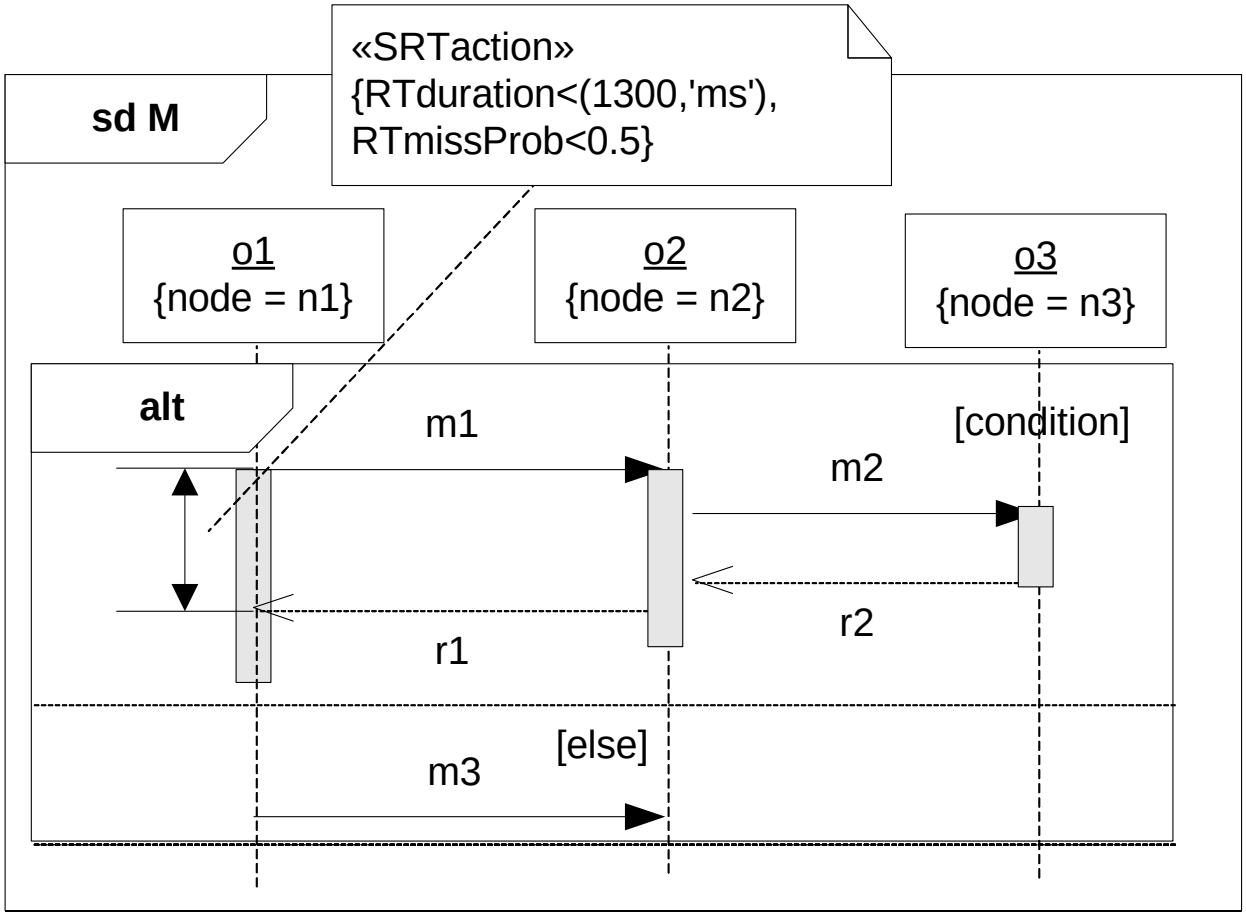
Vahid Garousi

- **Definitions:**
  - **Real-time (RT) fault: The execution time of a RT task has passed its RT constraint (RT deadline, e.g., 1 second)**
  - **Example: The anti-lock brakes on a car are a simple example of a RT computing system.**
  - **The RT constraint in this system is the time in which the brakes must be released to prevent the wheel from locking.**
  - **A RT deadline must be met, regardless of system load.**

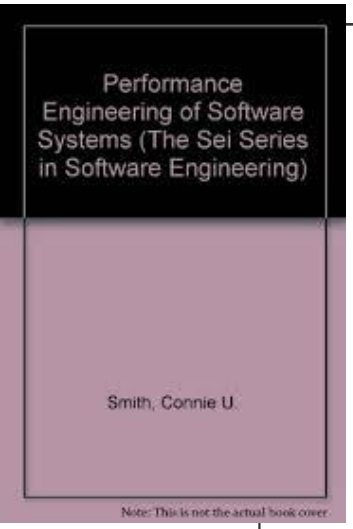


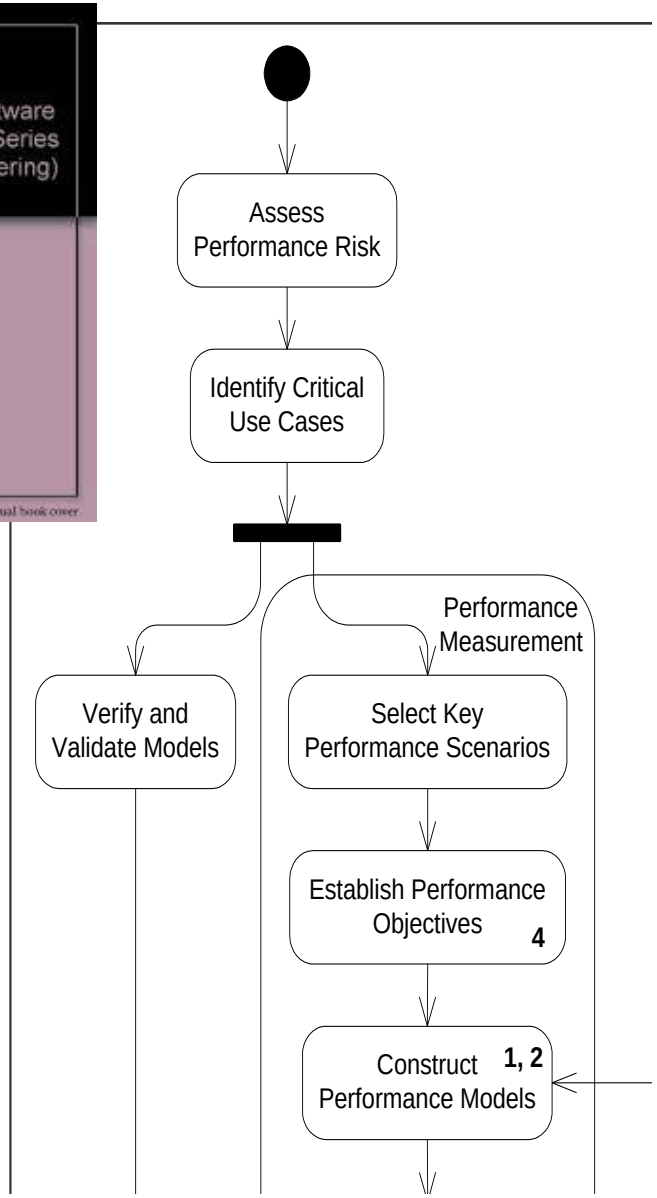
Modeling RT constraints in UML diagrams

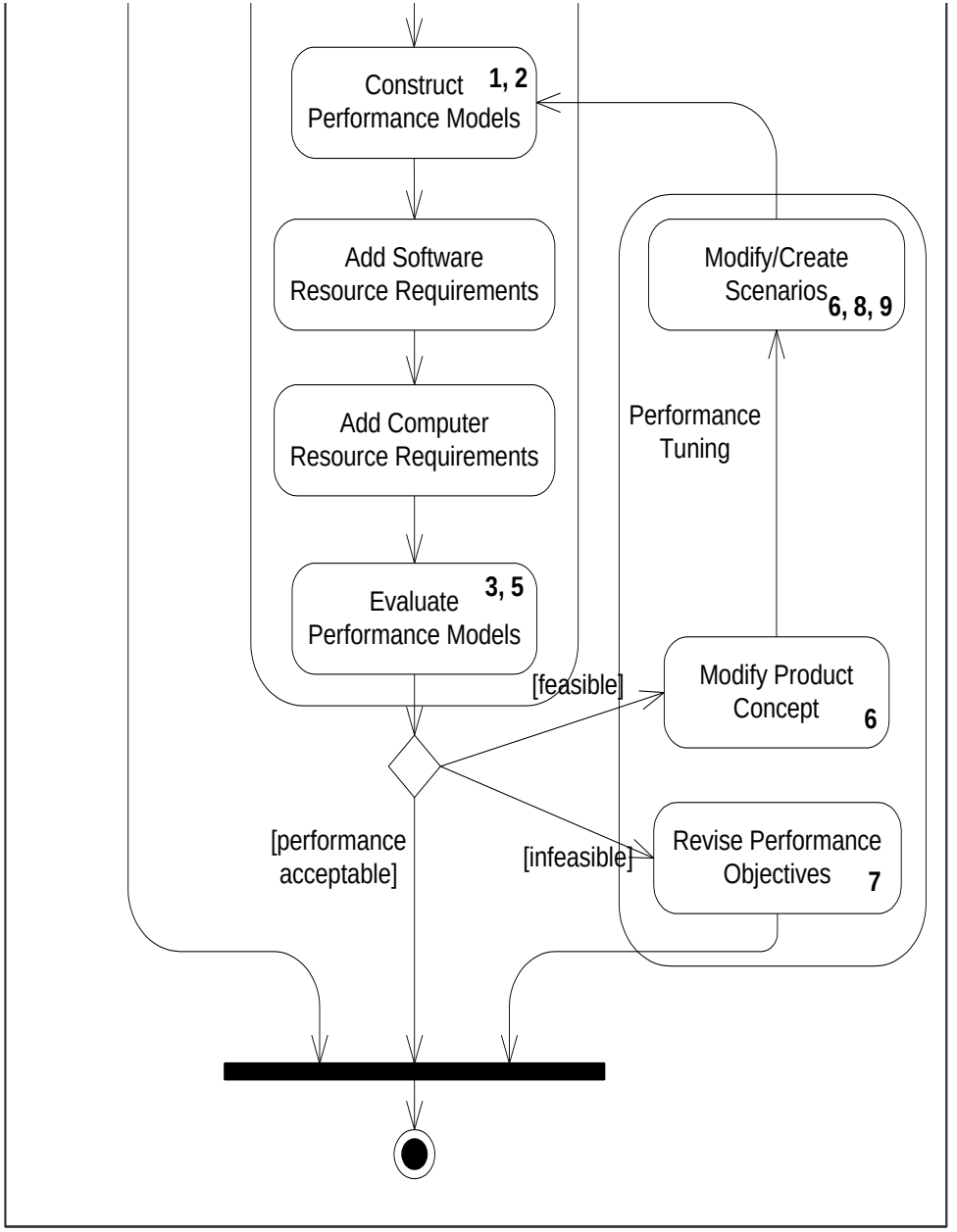
- **Usage of the «SRTaction» stereotype in a UML sequence diagram (soft versus hard RT actions)**
- **The UML Profile for MARTE: Modeling and Analysis of Real-Time and Embedded Systems**
  - www.omgmarte.org



An overview of the general SPE Process



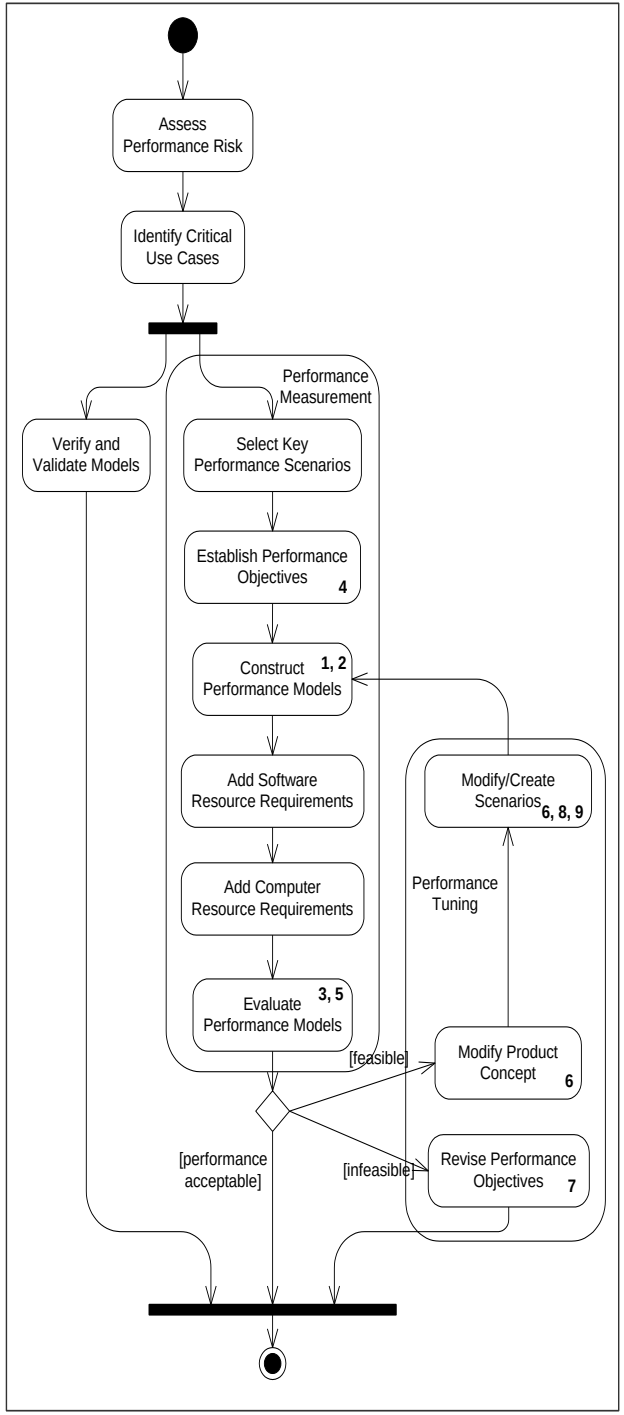




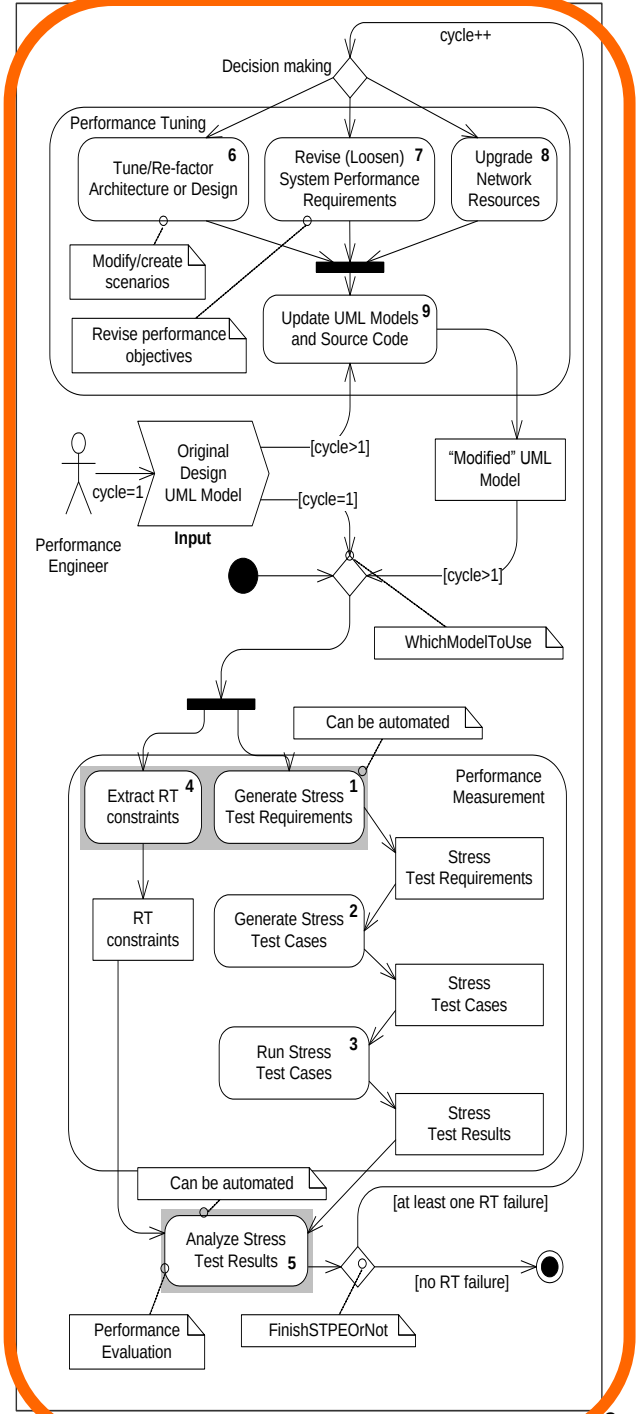
(a)-The General SPE Process

Our Stress-Test Performance Engineering (STPE) process versus the general SPE Process

Zoomed review next…

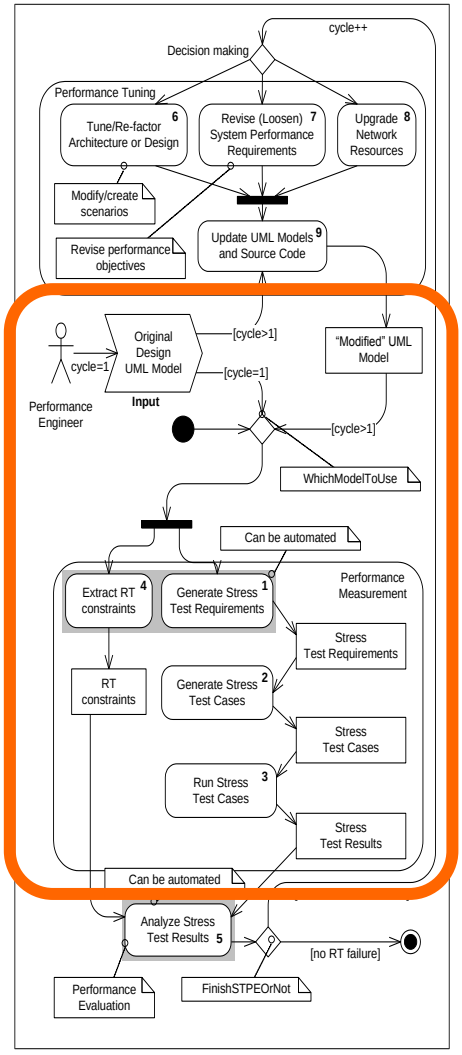


(a)-The General SPE Process

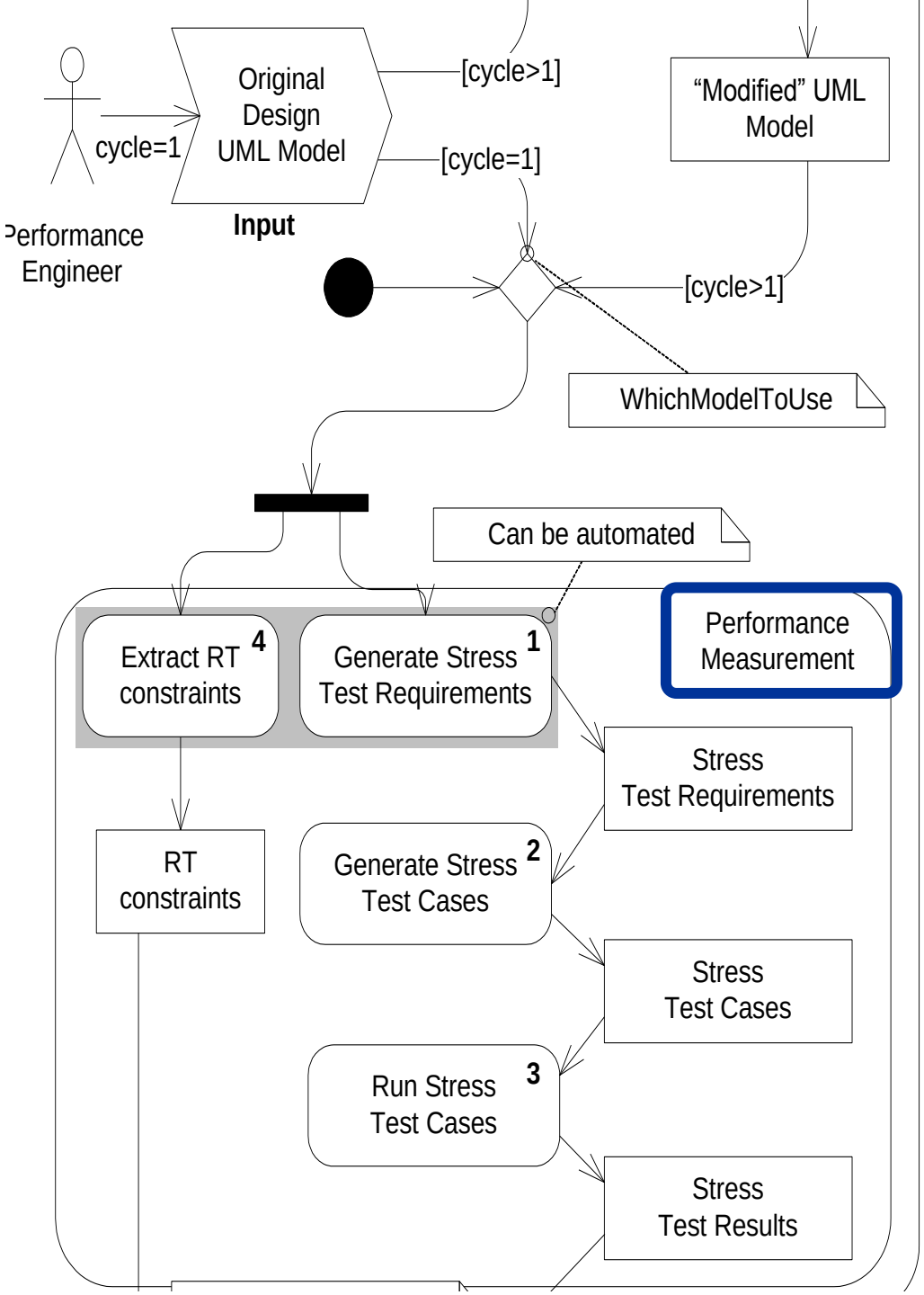


(b)-Stress-Test Performance Engineering Process

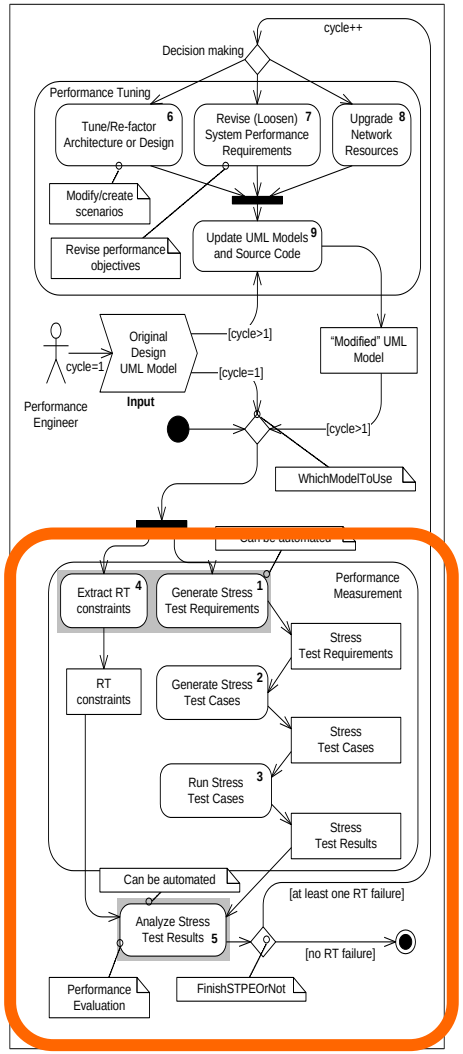
Our Stress-Test Performance Engineering (STPE) process



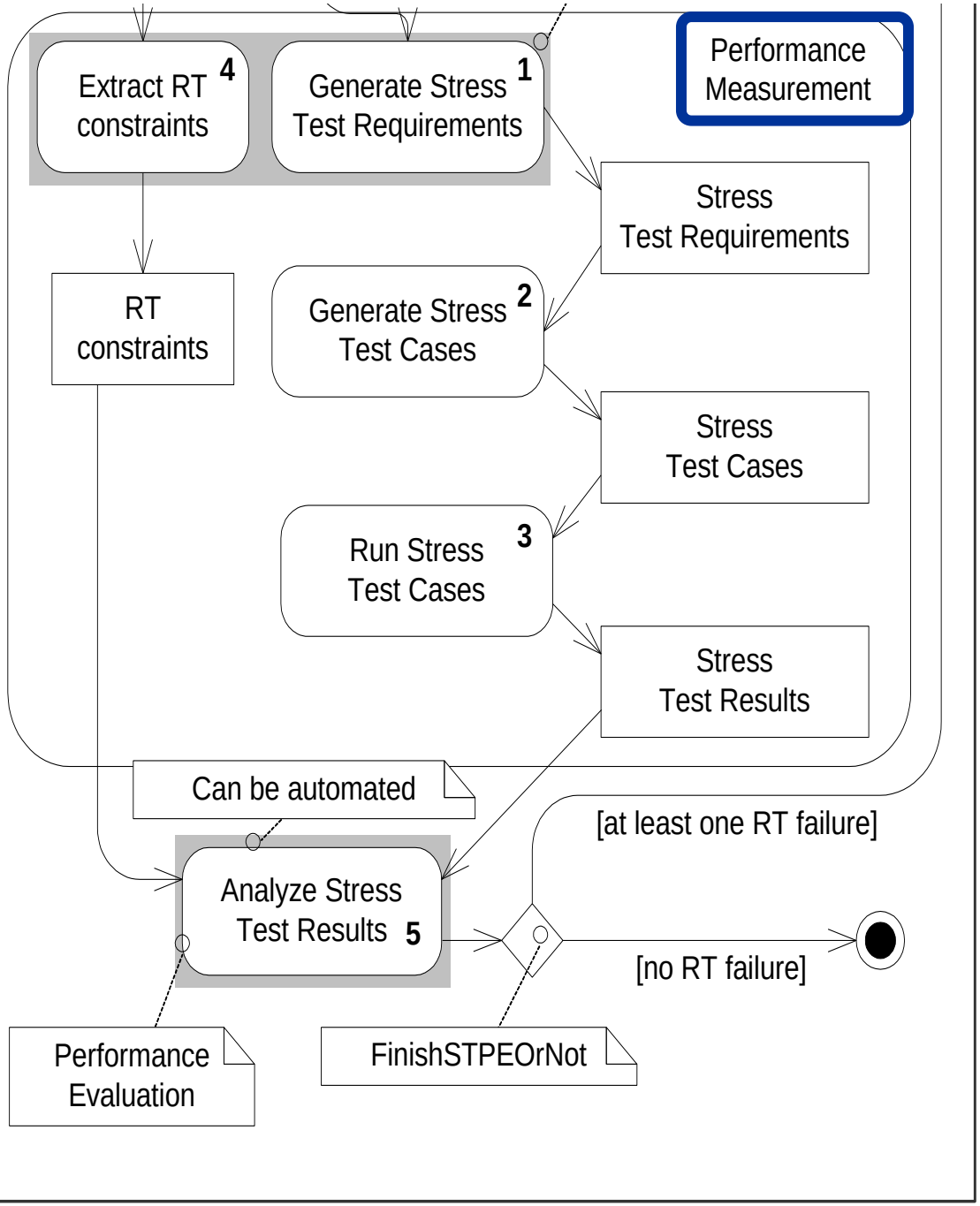
(b)-Stress-Test Performance Engineering Process



Our Stress-Test Performance Engineering (STPE) process

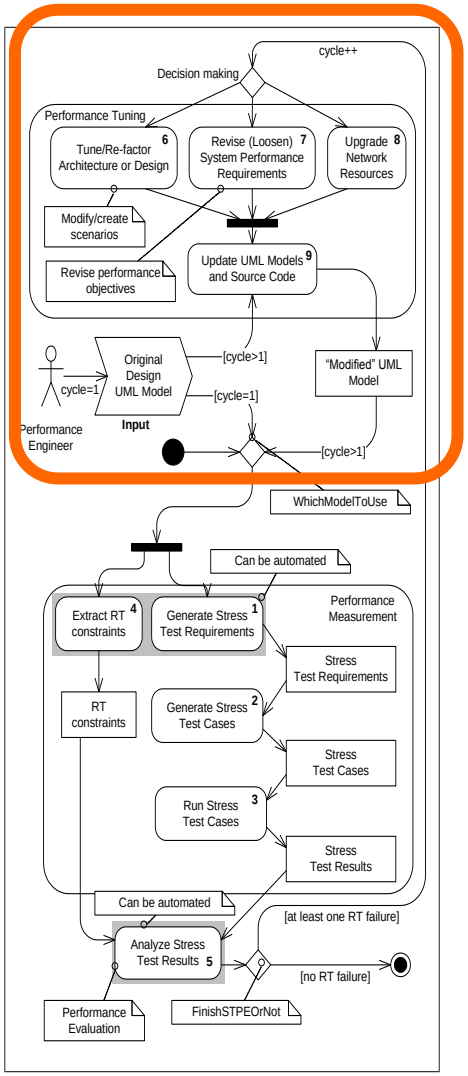


(b)-Stress-Test Performance Engineering Process



(b)-Stress-Test Performance Engineering Process

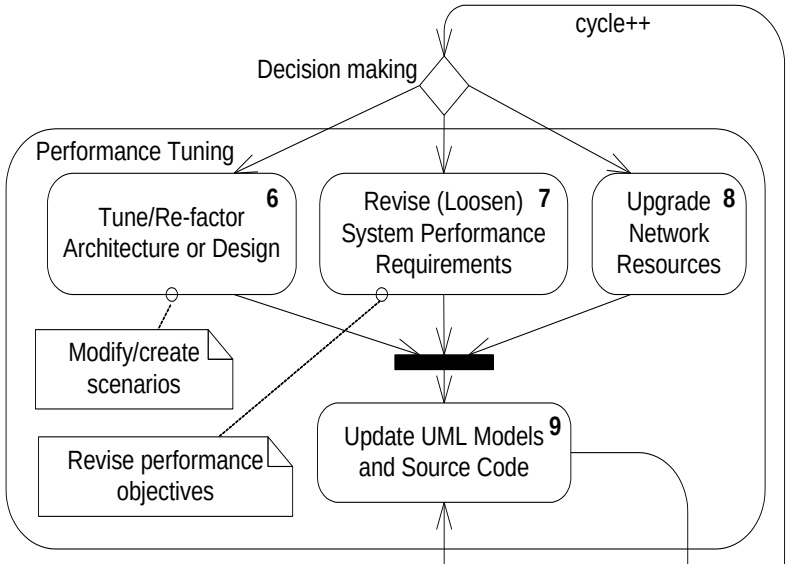
Our Stress-Test Performance Engineering (STPE) process

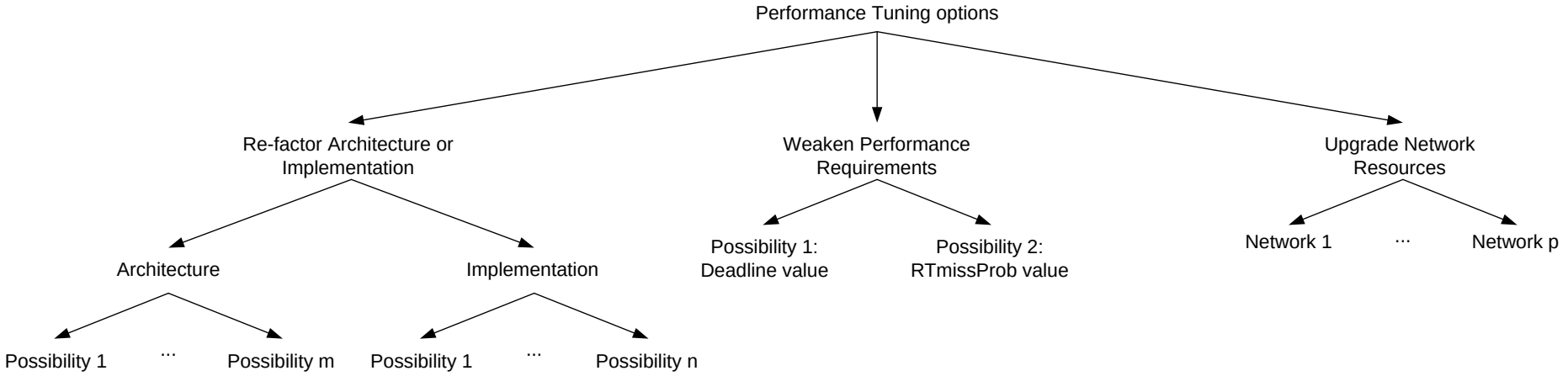


(b)-Stress-Test Performance Engineering Process

Performance tuning

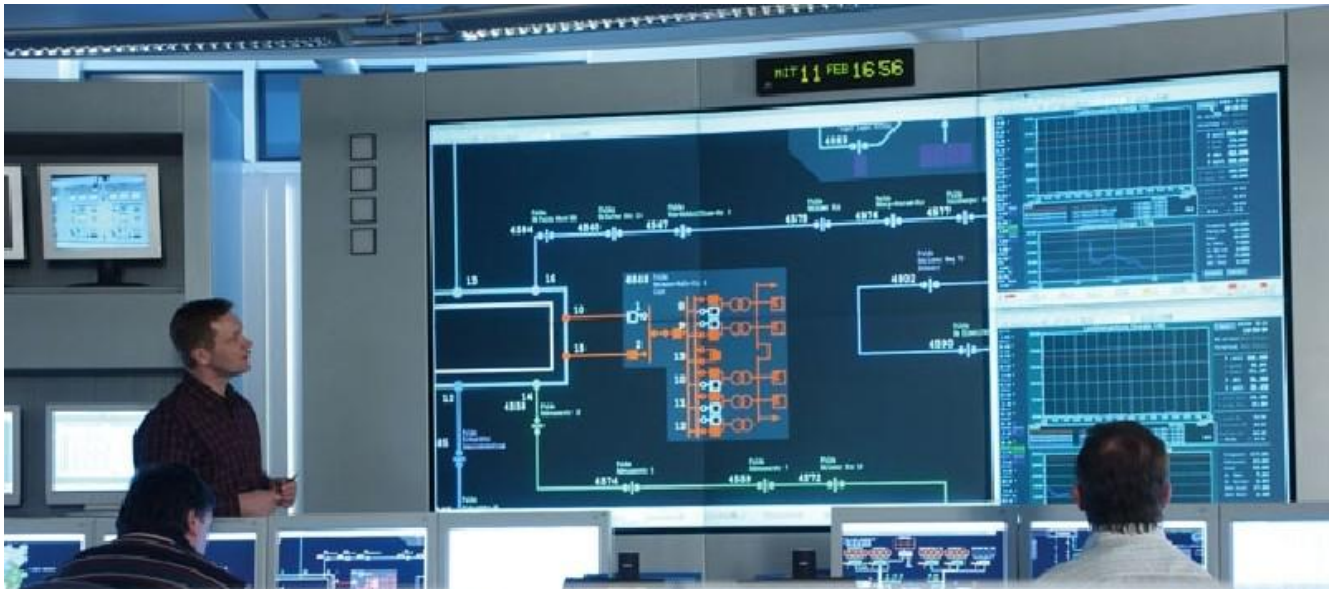
- **The performance tuning stage of STPE is undertaken if the performance evaluation in Step 5 reports that at least one RT failure (HRT or SRT) has occurred.**
- **A decision tree for Value-Based Performance Engineering (VBPE)…**
- **Just like Value-Based Software Engineering (VBSE)**





A case-study Experiment

- **System under analysis:**
  - A prototype SCADA-based power distribution system
  - SCADA: Supervisory Control And Data Acquisition
  - A system to control the power distribution grid across Canada consisting of several provinces.
  - Each province has several cities and regions.
  - There is one central server in each province which gathers the SCADA data from Tele-Control units (TCs) from all over the province and sends them to the national server.

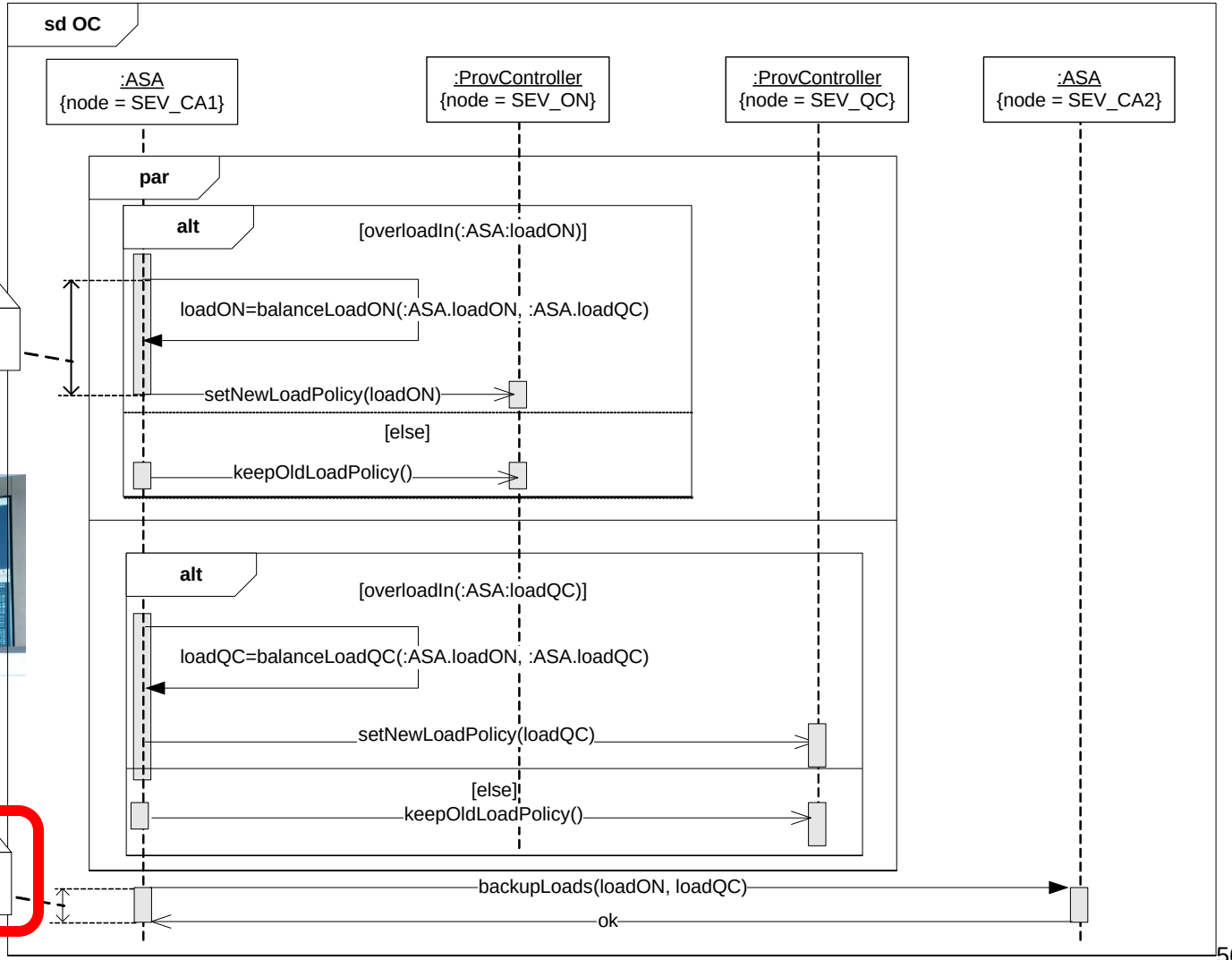


hydro one

SIEMENS

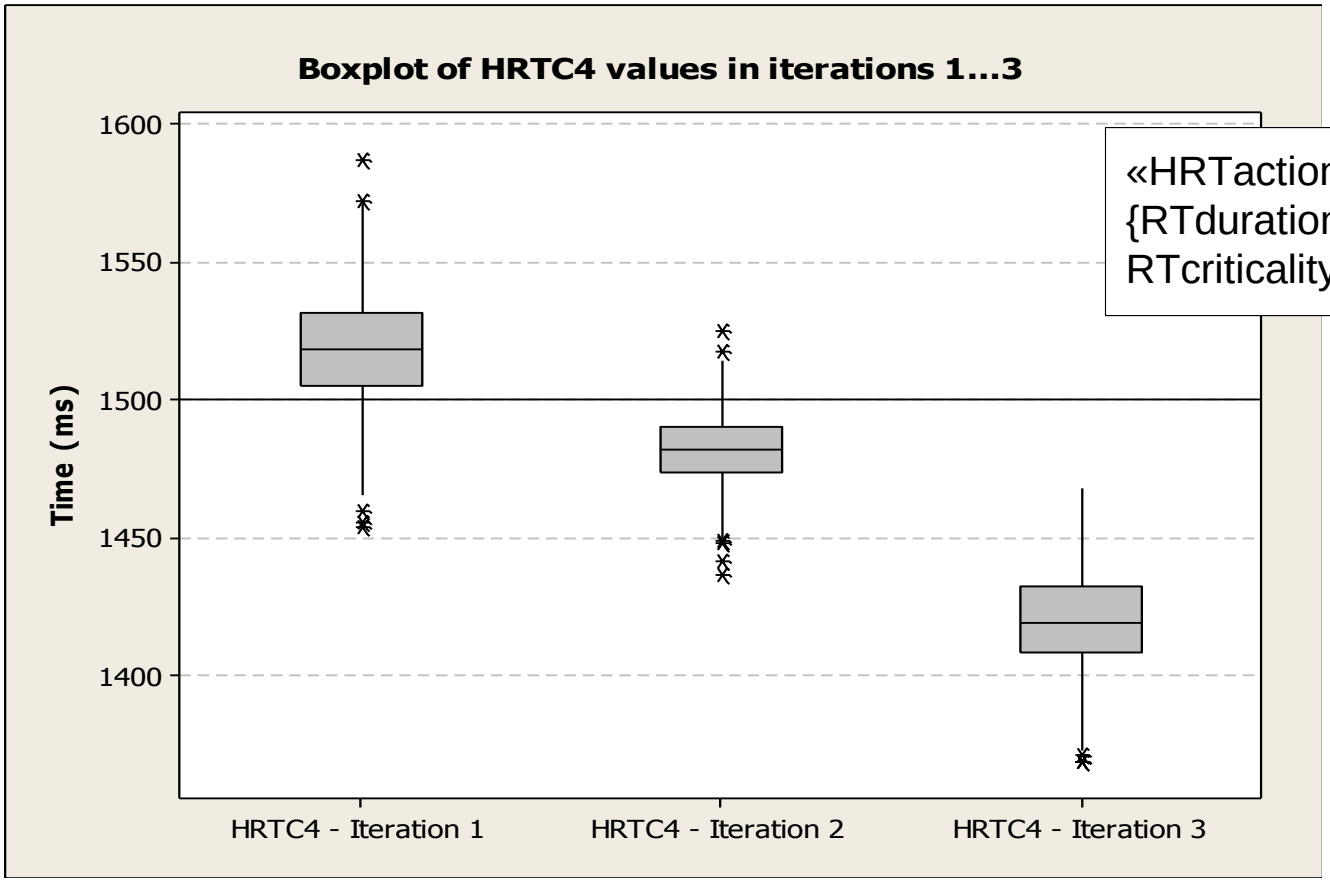
A case-study Experiment

- **RT constraints of the system under analysis**
- **Overload Control (OC) use-case of two Canadian provinces: Ontario and Quebec**



A case-study Experiment

- **Applying the Stress-Test Performance Engineering (STPE)**
  - Iteration 1: we conducted architecture/design refactoring
  - Iteration 2: we replaced the wireless network card (speed: 19 Mbps) of *SEV_CA1* with a faster wired network cards (speed: 100 Mbps).



«HRTaction» **$HRTC_4$**
{RTduration<(1500,'ms'),
RTcriticality=0.95}

Outline

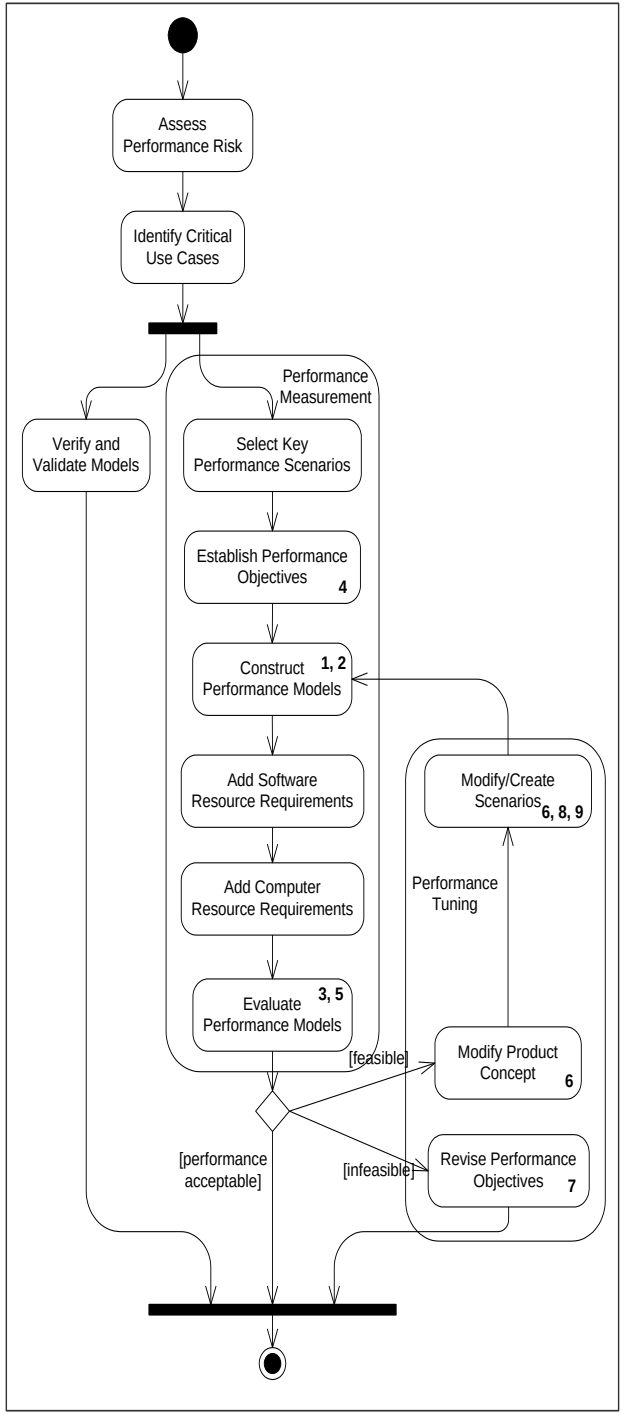
- Background of the speaker and his research expertise
- A systematic mapping of UML-Driven Software Performance Engineering (UML-SPE)
- Review of one UML-SPE technique

- My team and I would be glad to collaborate with other researchers and practitioners… ☺

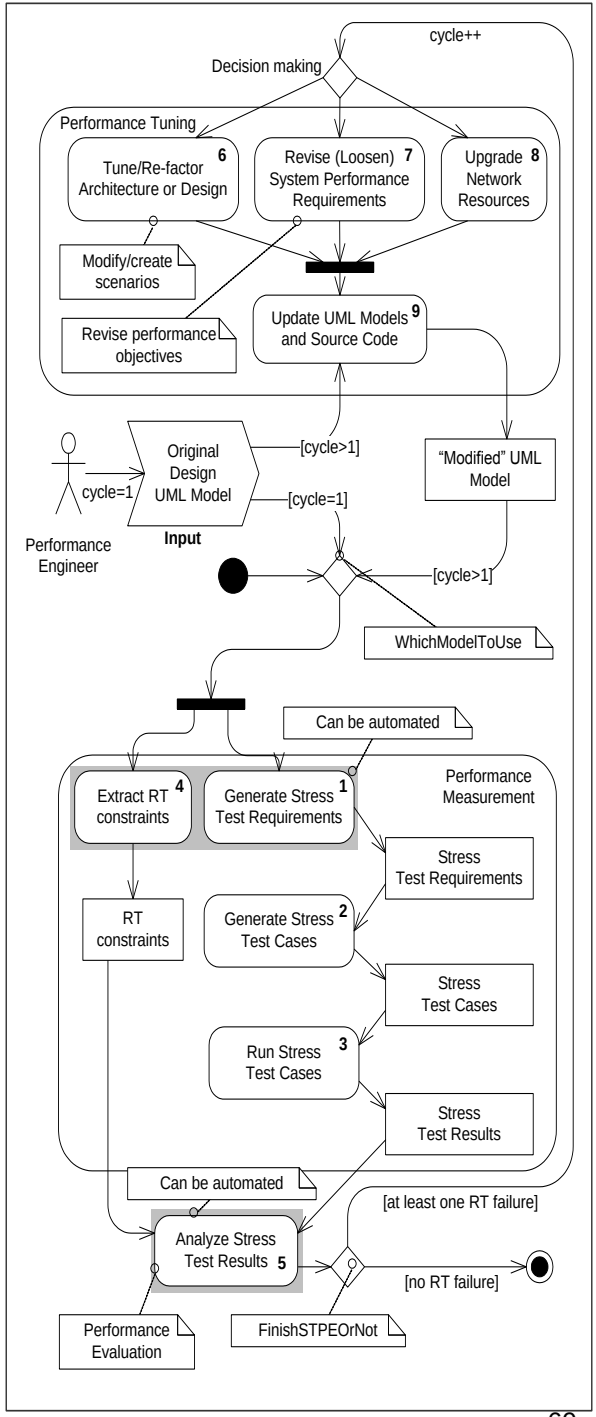
Q/A

BACK UP

Our Stress-Test Performance Engineering (STPE) process versus the general SPE Process



(a)-The General SPE Process



(b)-Stress-Test Performance Engineering Process