Live Trace Visualization for System and Program Comprehension in Large Software Landscapes

Seminar @ Lugano

Kiel University, Software Engineering Group

Florian Fittkau — July 1, 2015

ExplorViz

Motivation

- Increasing code complexity on application level
- Likewise increase of systems and communications (e.g., microservices)
- Systems often form a large software landscape
- Knowledge about usage and communication of each system often gets lost

Live trace visualization of those landscapes for comprehension of systems and applications

ExplorViz

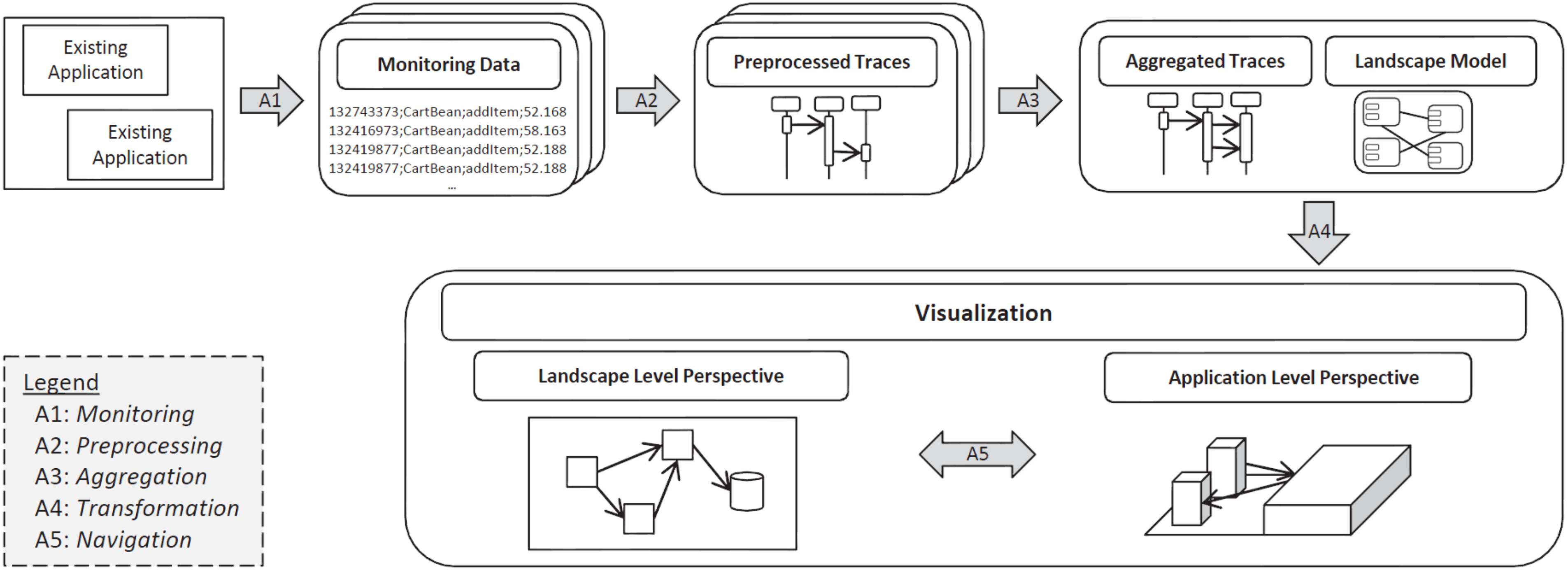
Selected Challenges:

- Possible huge monitoring data amount (performance/cost efficiency)
- Finding abstractions to understand huge landscapes but also application-level details
- Live visualization of thousands or even millions of traces

[Fittkau et al. 2013]

Approach & Contributions

The ExplorViz Method



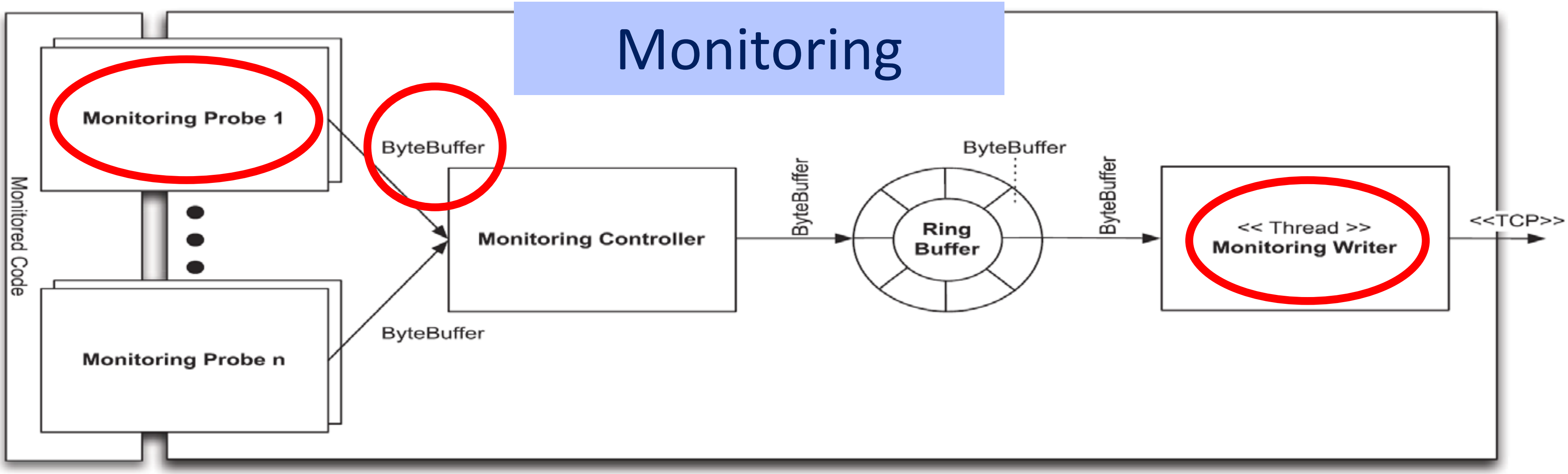
```
132743373;CartBean;addItem;52.168
132416973;CartBean;addItem;58.163
132419877;CartBean;addItem;52.188
132419877;CartBean;addItem;52.188
...
```

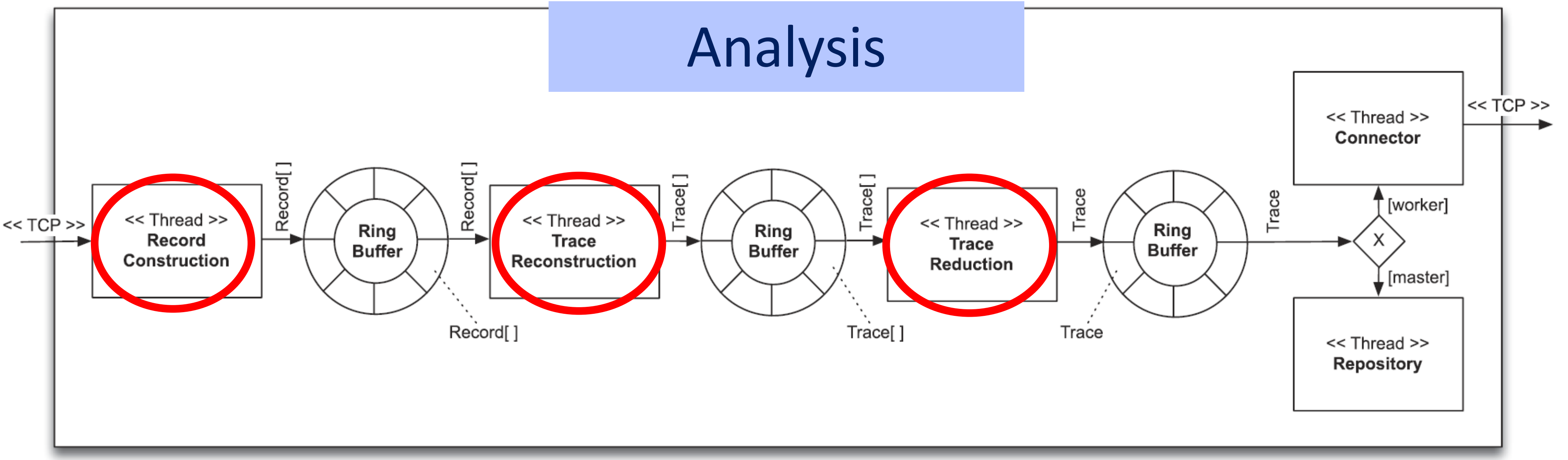
Legend

A1: *Monitoring*
A2: *Preprocessing*
A3: *Aggregation*
A4: *Transformation*
A5: *Navigation*

[Fittkau et al. 2013a]

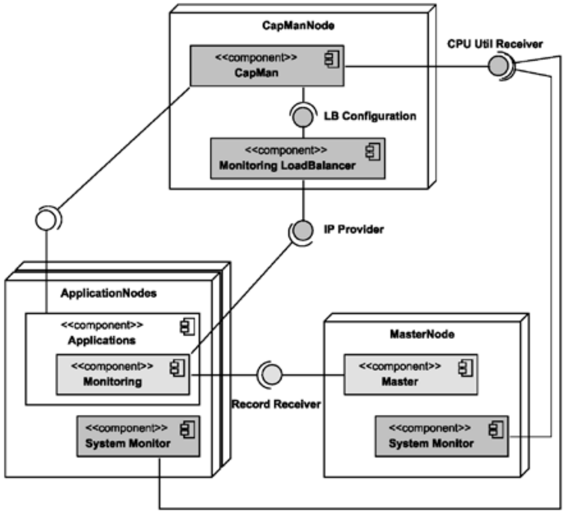
Monitoring & Trace Processing

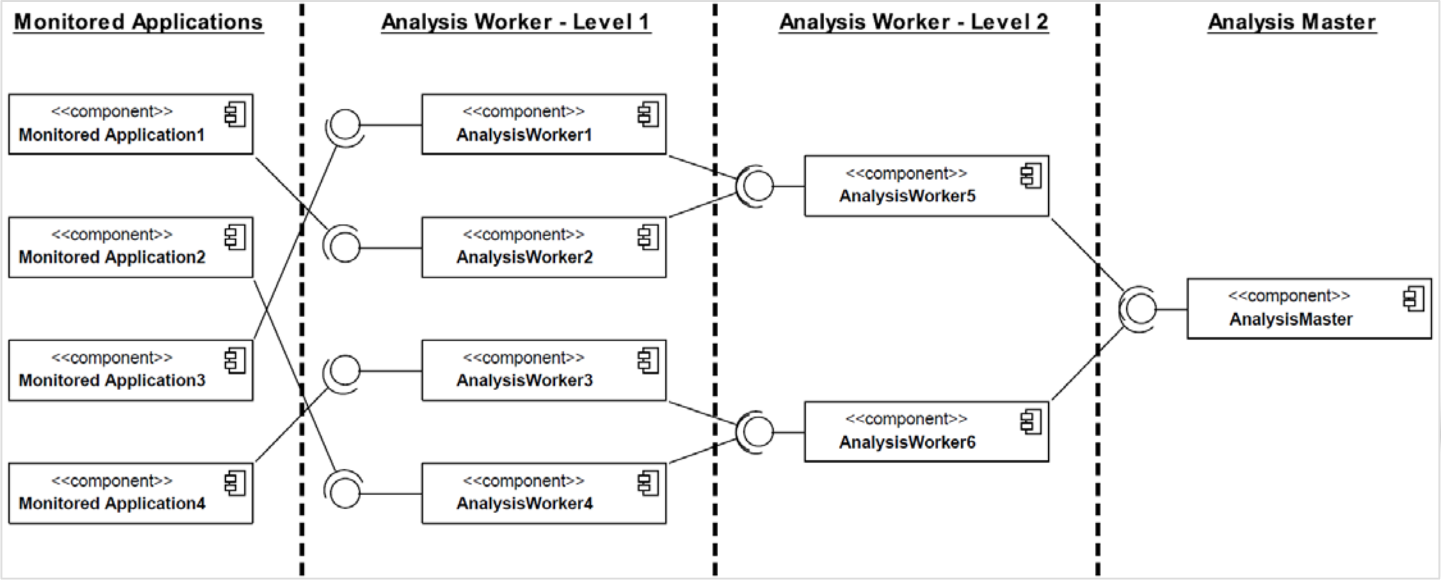




[Fittkau et al. 2013b, Beye 2013, Matthiessen 2014, Weißenfels 2014]

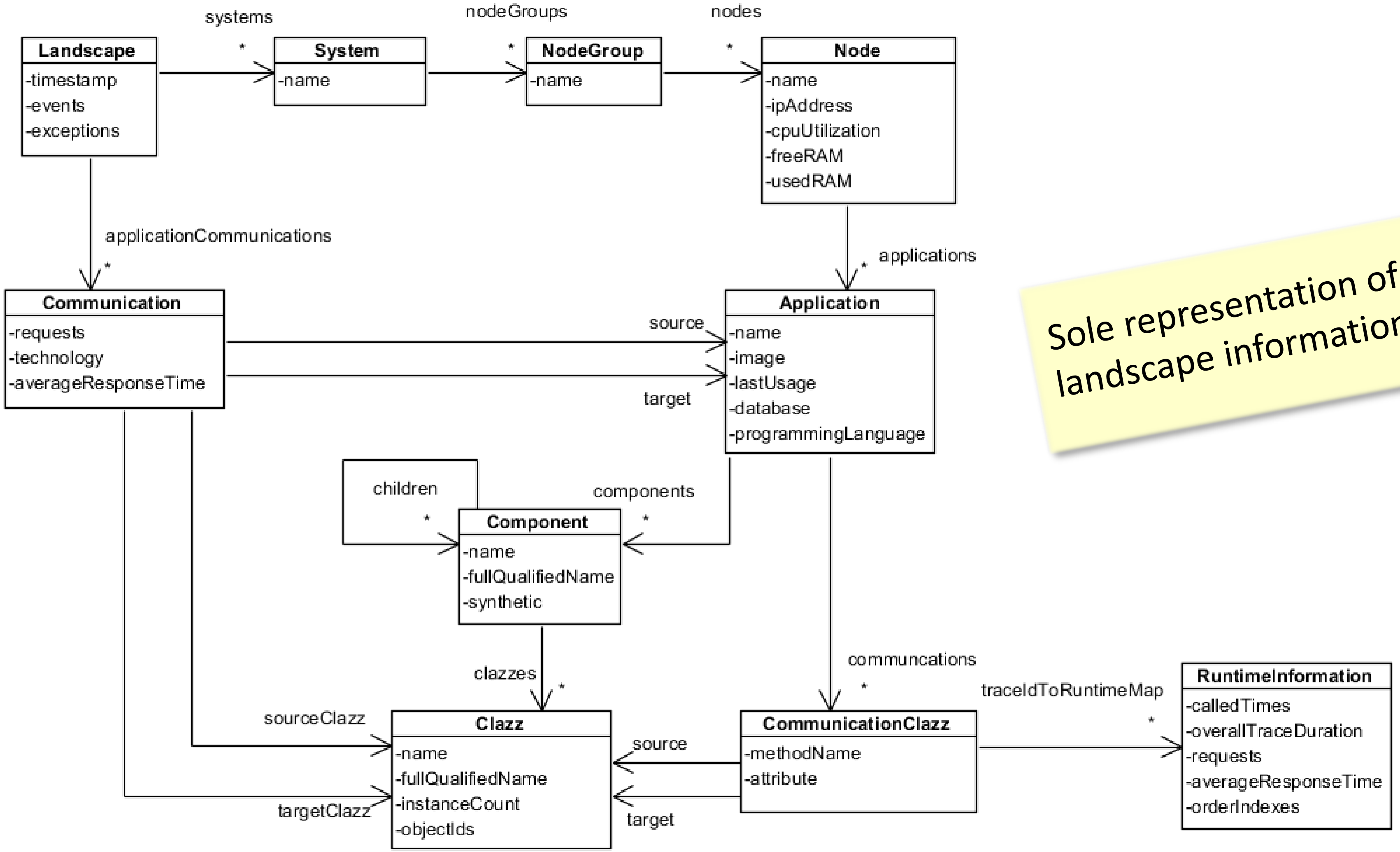
Scalable Trace Processing





[Fittkau et al. 2014b, Koppenhagen 2013, Stelzer 2014]

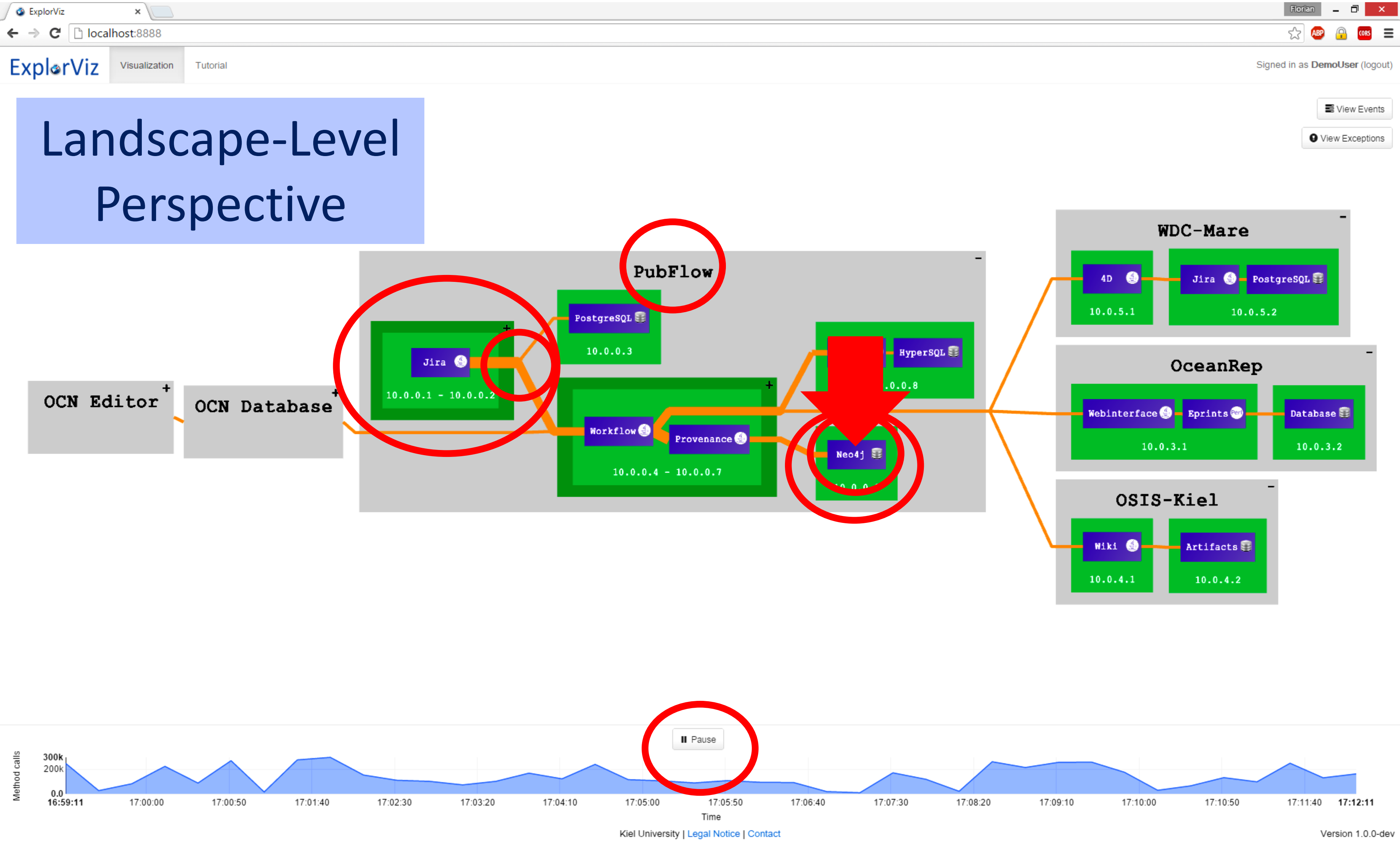
Landscape Meta-Model



Sole representation of landscape information

ExplorViz Visualization

Christian-Albrechts-Universität zu Kiel

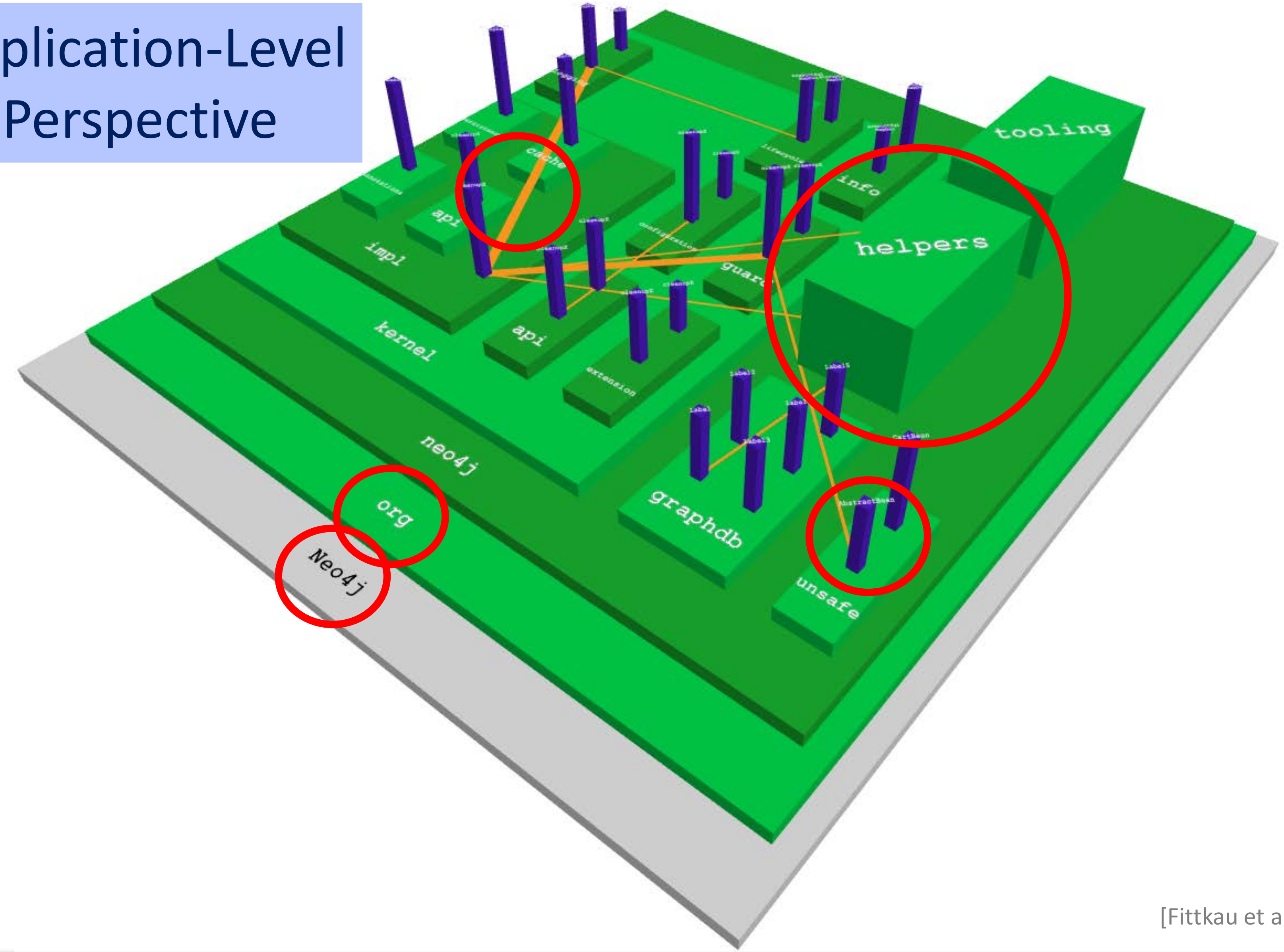


Landscape-Level Perspective

[Fittkau et al. 2013a]

ExplorViz Visualization

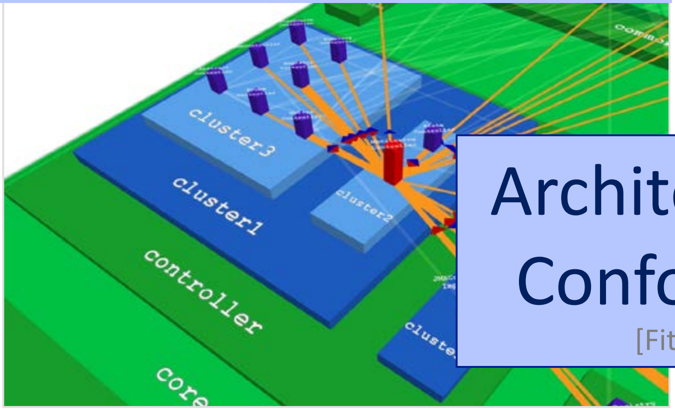
Application-Level Perspective



[Fittkau et al. 2013a]

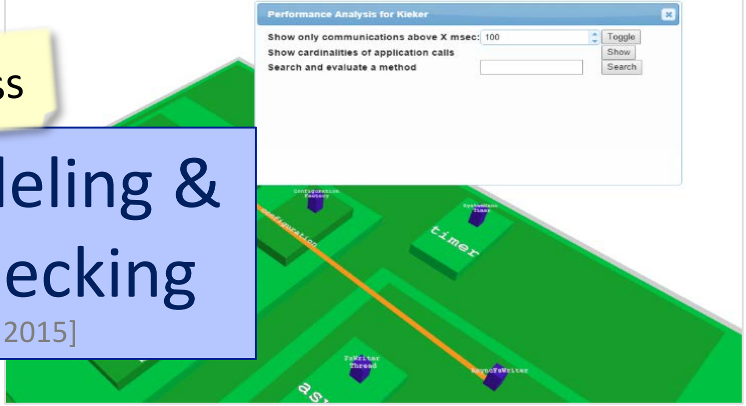
Selected ExplorViz Features

Clustering



[Barzel 2014]

Performance Analysis


Performance Analysis for Kieker

Show only communications above X msec: 100 Toggle

Show cardinalities of application calls Show

Search and evaluate a method Search

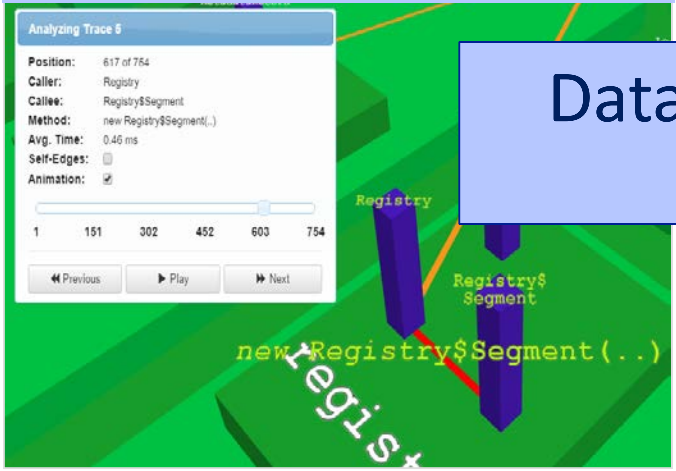
[Jähde 2015]

Work in Progress

Architecture Modeling & Conformance Checking

[Fittkau et al. 2014b, Simolka 2015]

Trace Replayer


Analyzing Trace 6

Position: 617 of 754

Caller: Registry

Callee: Registry$Segment

Method: new Registry$Segment(..)

Avg. Time: 0.46 ms

Self-Edges: □

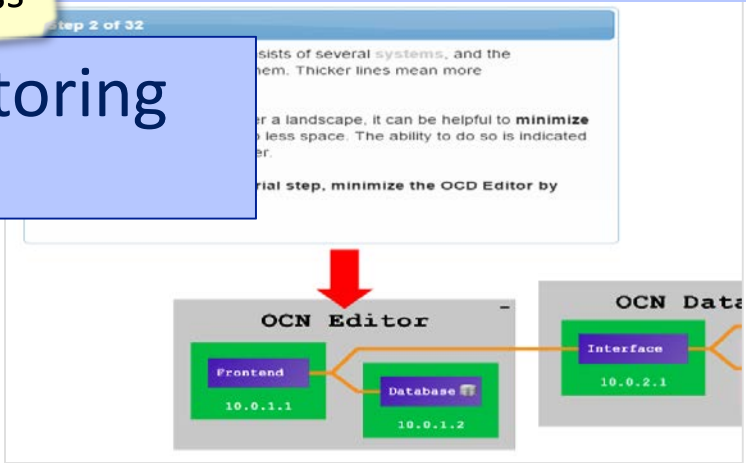
Animation: ☑

Work in Progress

Database Monitoring

[Zirkelbach 2015]

Interactive Tutorial


Step 2 of 32

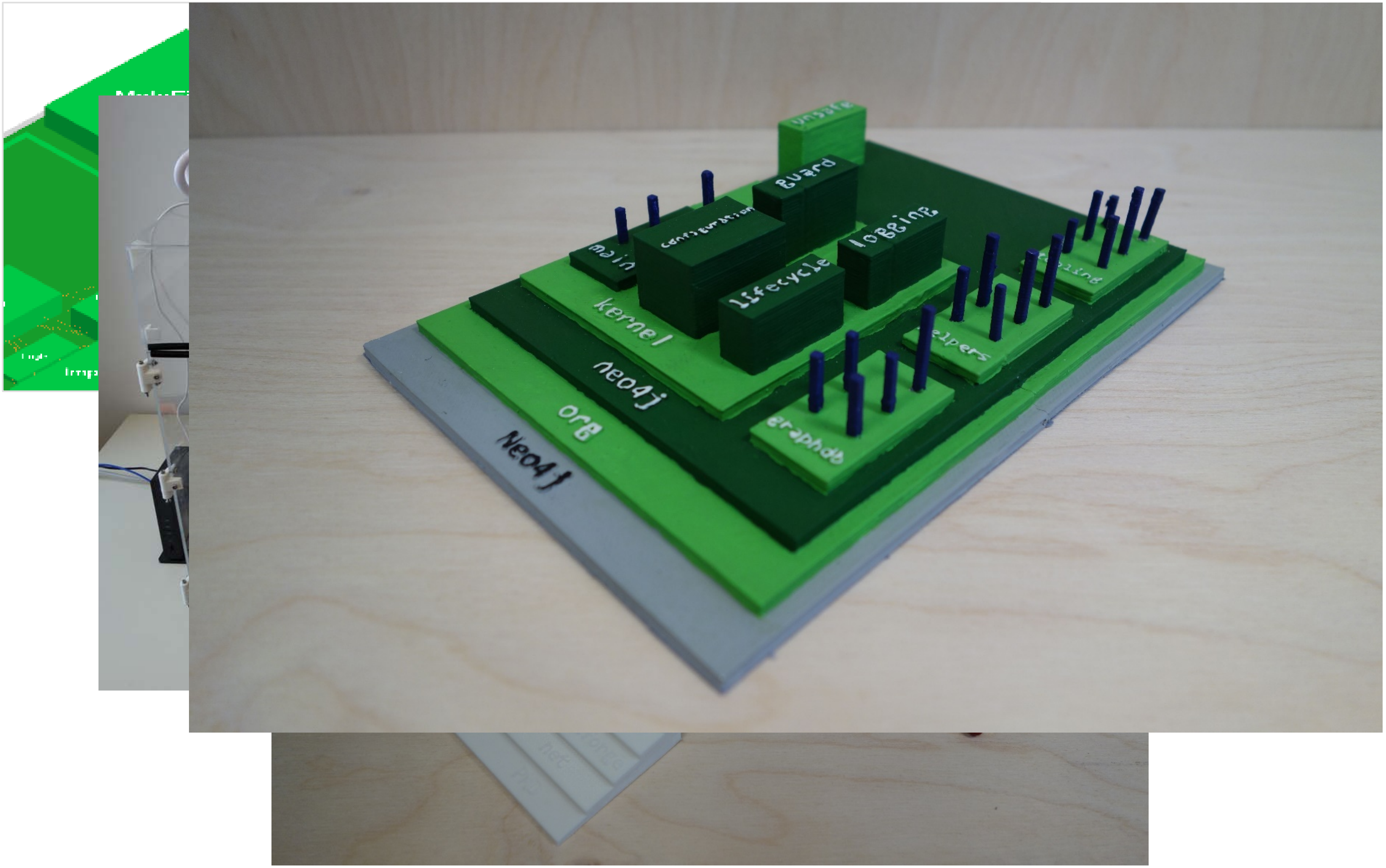
sists of several systems, and the
hem. Thicker lines mean more

er a landscape, it can be helpful to **minimize**
less space. The ability to do so is indicated
er.

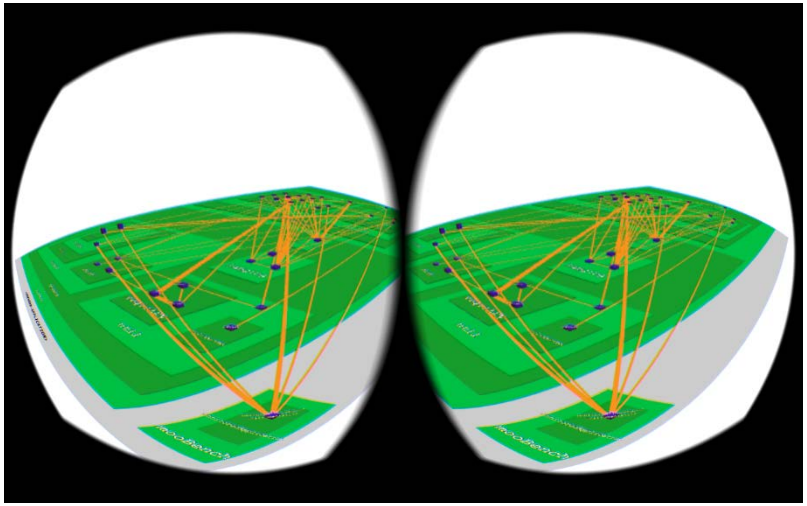
rial step, minimize the OCD Editor by

[Finke 2014]

Physical 3D Models



Virtual Reality



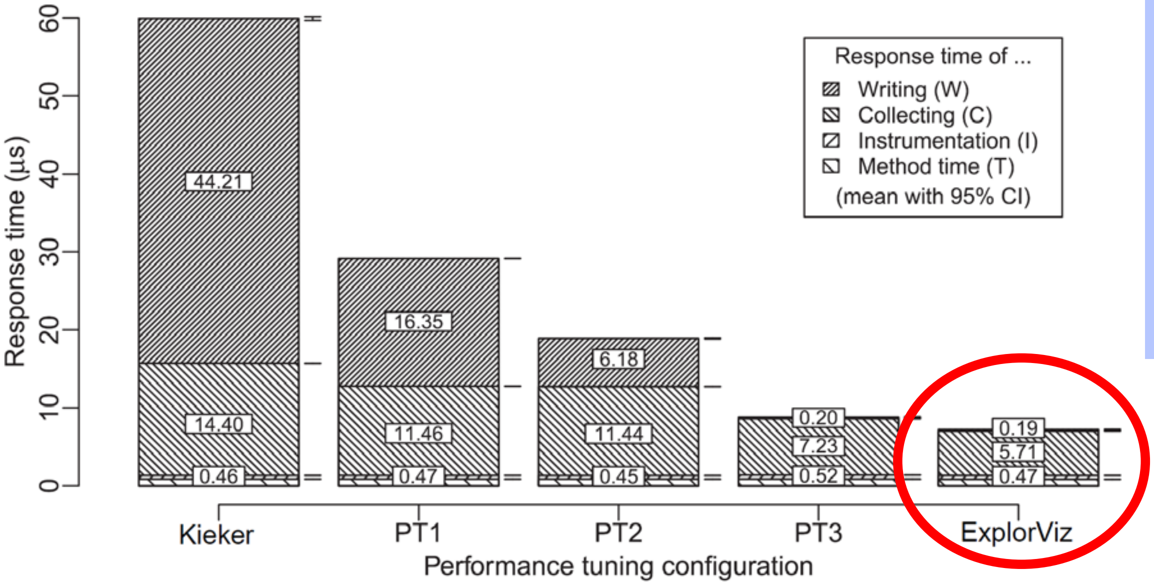
[Krause 2015]

Virtual Reality

[Krause 2015]

Evaluations

Monitoring & Trace Processing



Kieker 1.8:

16.6k traces per second

ExplorViz:

141.2k traces per second

Speedup of about **factor 9**

[Waller et al. 2014]

Kieker 1.8 with Analysis:

0.5k traces per second

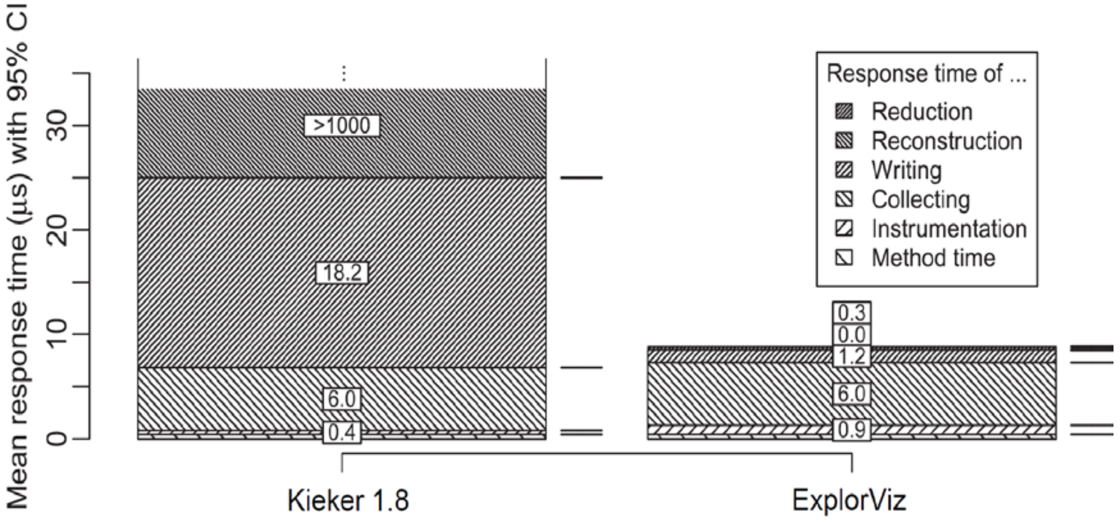
ExplorViz with Analysis:

112.6k traces per second

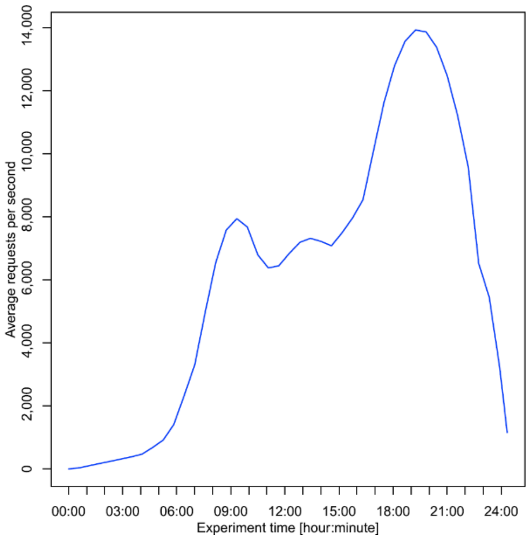
(limited by network speed)

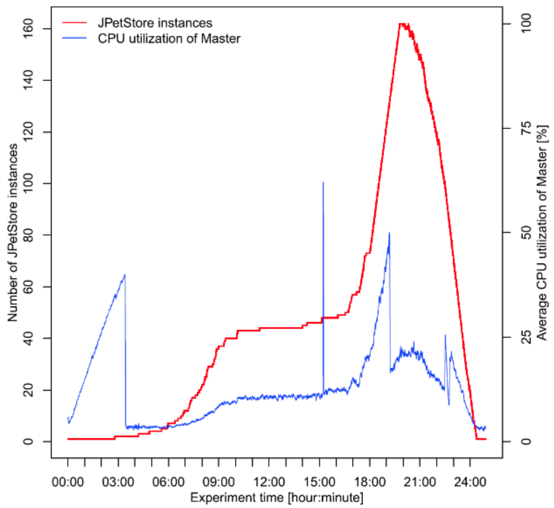
Speedup of about **factor 250**

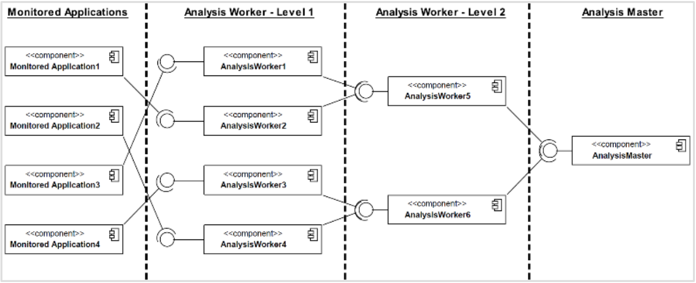
[Fittkau et al. 2013b]

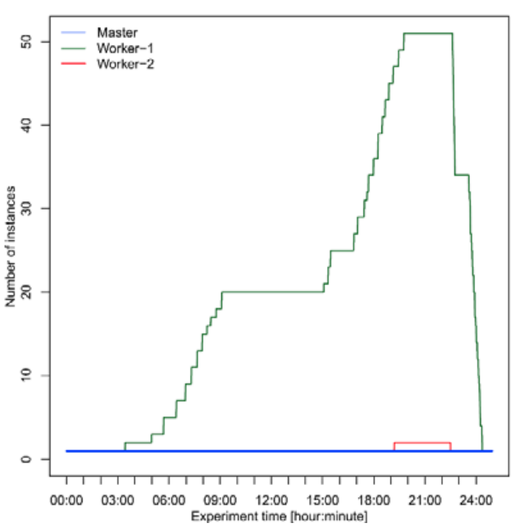


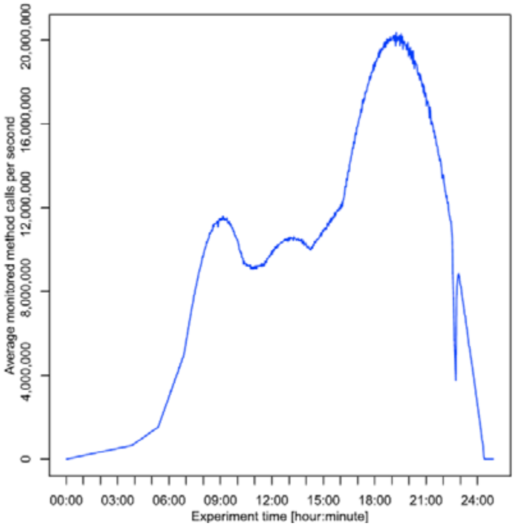
Scalable Trace Processing






Monitored Applications
Analysis Worker - Level 1
Analysis Worker - Level 2
Analysis Master
<<component>> Monitored Application1
<<component>> Monitored Application2
<<component>> Monitored Application3
<<component>> Monitored Application4
<<component>> AnalysisWorker1
<<component>> AnalysisWorker2
<<component>> AnalysisWorker3
<<component>> AnalysisWorker4
<<component>> AnalysisWorker5
<<component>> AnalysisWorker6
<<component>> AnalysisMaster





Processing Capabilities:

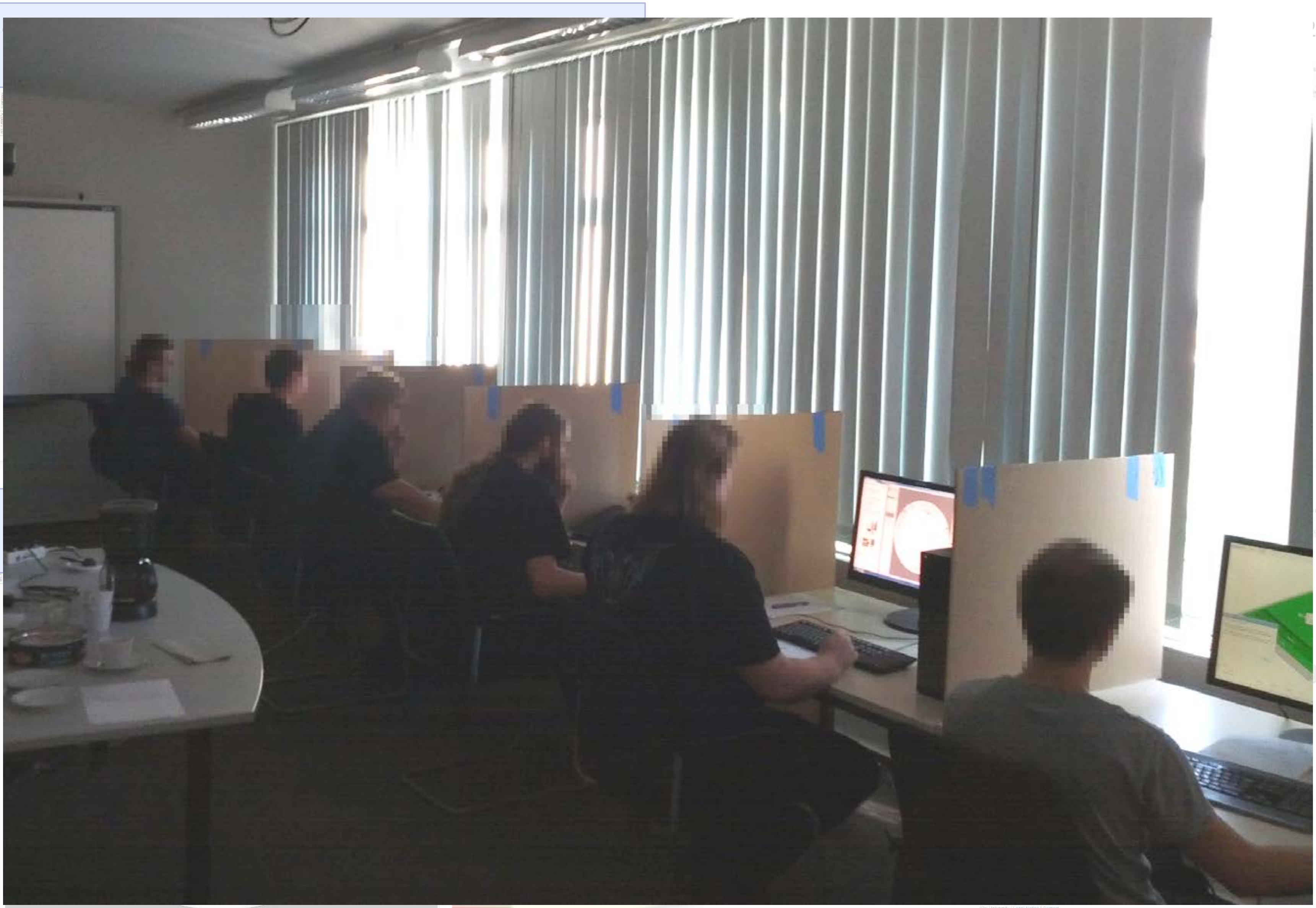
- ✓ Cost efficient
- ✓ Scalable to millions of monitored methods

[Fittkau et al. 2014b, Stelzer 2014]

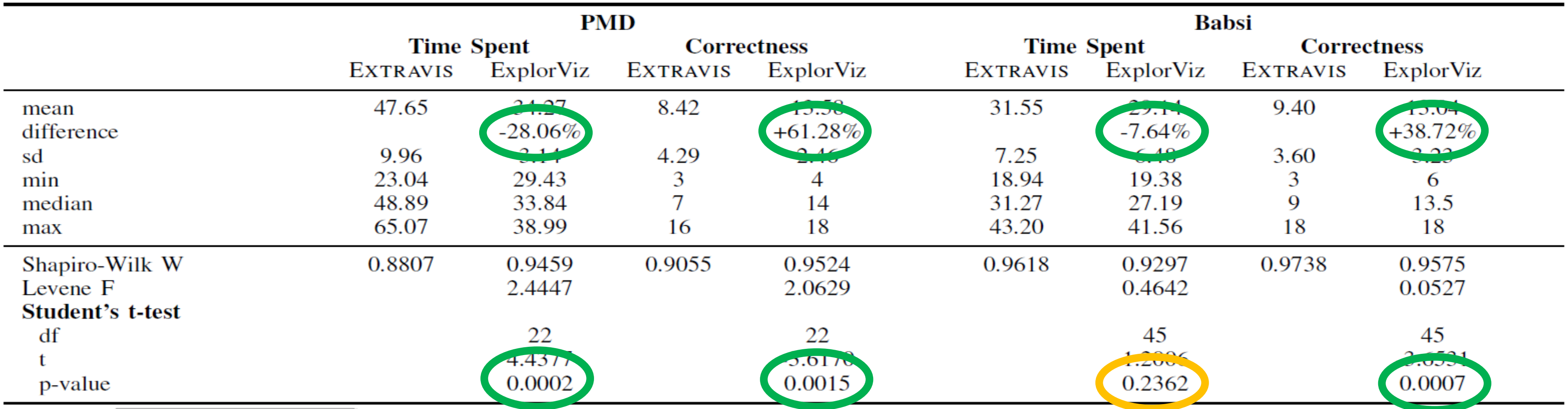
Application Perspective

F

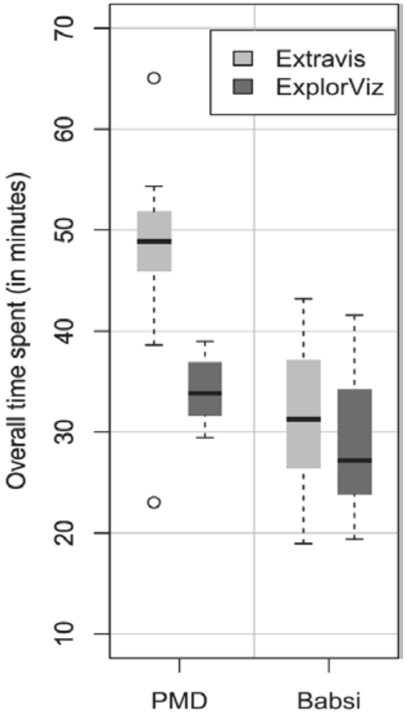
2013]

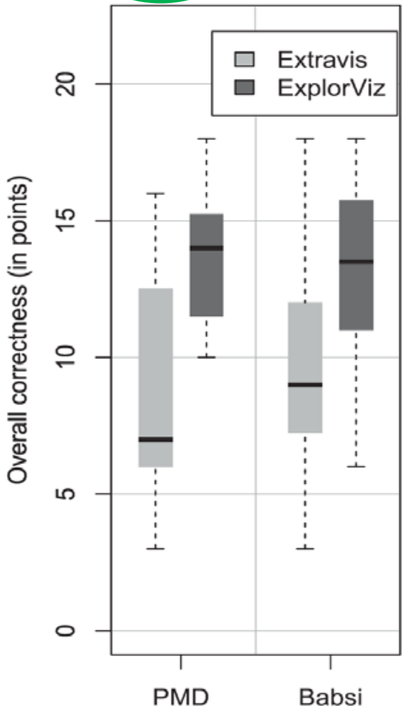


Application Perspective



	PMD				Babsi			
	Time Spent		Correctness		Time Spent		Correctness	
	EXTRAVIS	ExplorViz	EXTRAVIS	ExplorViz	EXTRAVIS	ExplorViz	EXTRAVIS	ExplorViz
mean	47.65	34.27	8.42	13.58	31.55	29.14	9.40	13.04
difference		-28.06%		+61.28%		-7.64%		+38.72%
sd	9.96	3.14	4.29	2.46	7.25	6.48	3.60	3.23
min	23.04	29.43	3	4	18.94	19.38	3	6
median	48.89	33.84	7	14	31.27	27.19	9	13.5
max	65.07	38.99	16	18	43.20	41.56	18	18
Shapiro-Wilk W	0.8807	0.9459	0.9055	0.9524	0.9618	0.9297	0.9738	0.9575
Levene F		2.4447		2.0629		0.4642		0.0527
Student's t-test								
df		22		22		45		45
t		4.4377		3.6170		1.2006		3.6531
p-value		0.0002		0.0015		0.2362		0.0007

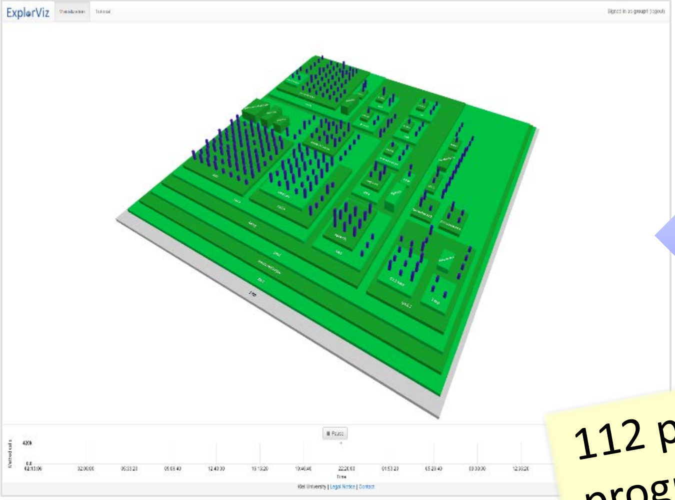




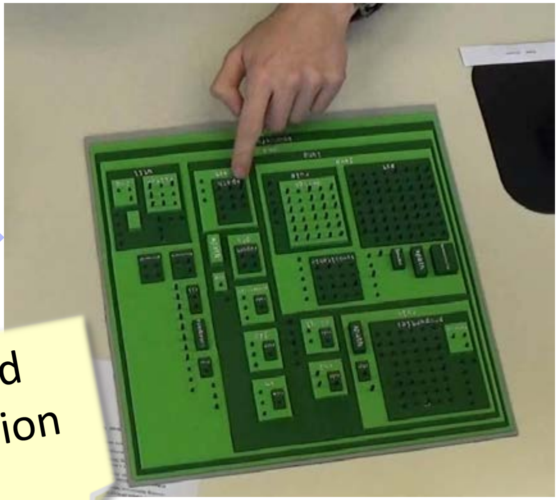
Visualization Capabilities:

- ✓ Usable for solving program comprehension tasks
- ✓ Interactive concept for method calls works
- ✓ More efficient and effective than competitor (for the chosen tasks)

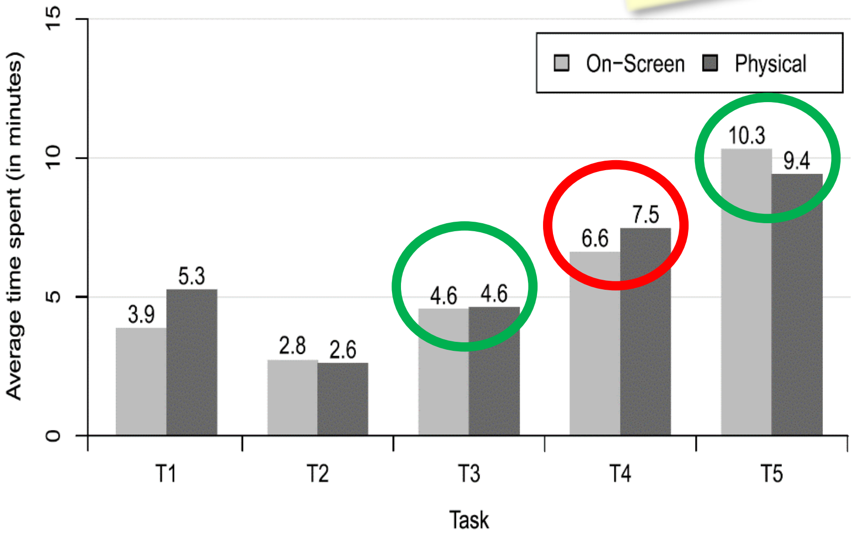
Physical 3D Model

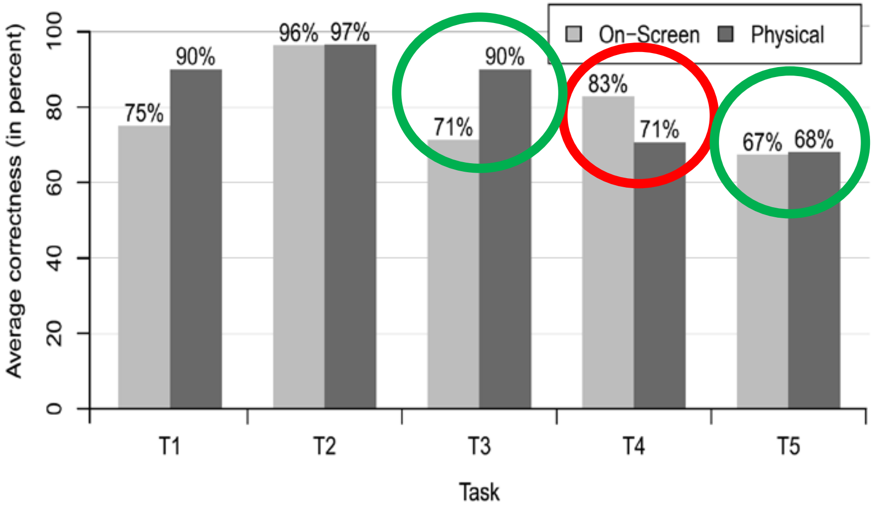


VS

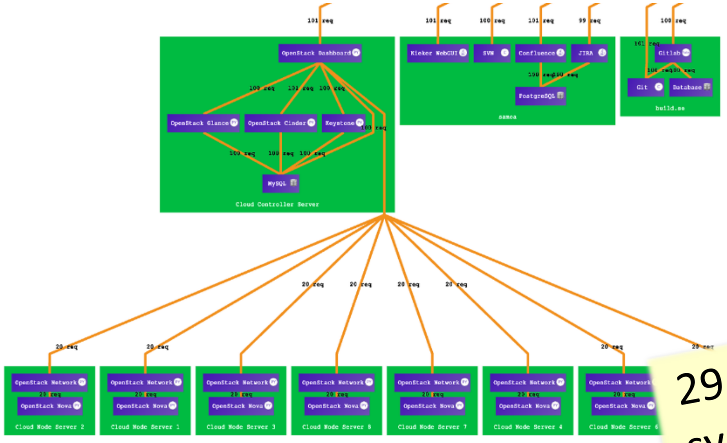


112 participants solved program comprehension tasks in pairs



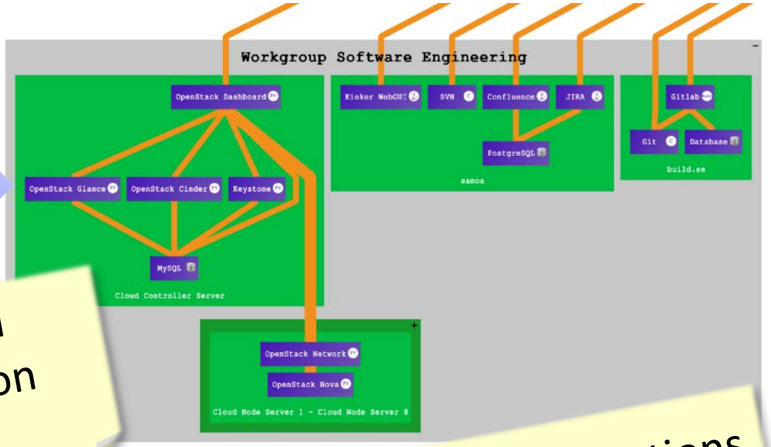


Landscape Perspective



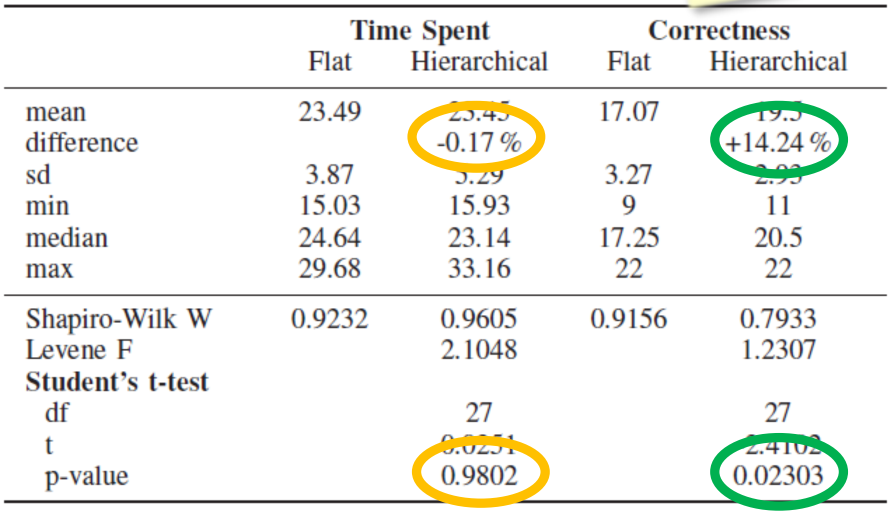
(State-of-the-art APM)

VS



29 participants solved system comprehension tasks

140 applications



	Time Spent		Correctness	
	Flat	Hierarchical	Flat	Hierarchical
mean	23.49	23.45	17.07	19.5
difference		-0.17 %		+14.24 %
sd	3.87	3.29	3.27	2.93
min	15.03	15.93	9	11
median	24.64	23.14	17.25	20.5
max	29.68	33.16	22	22
Shapiro-Wilk W	0.9232	0.9605	0.9156	0.7933
Levene F		2.1048		1.2307
Student's t-test				
df		27		27
t		0.0251		2.4102
p-value		0.9802		0.02303

Visualization Capabilities:

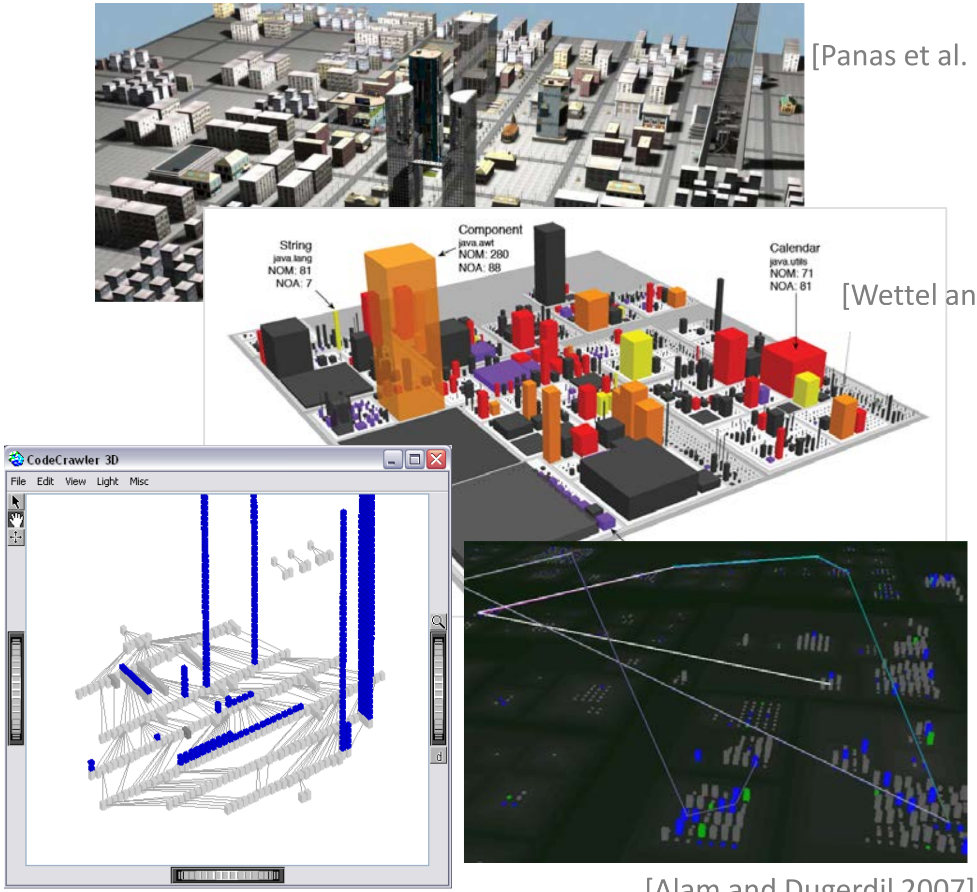
✓ Usable for system comprehension

✓ Abstractions provide a valuable addition

Related Work & Outlook

Closely Related Work

Software Visualizations



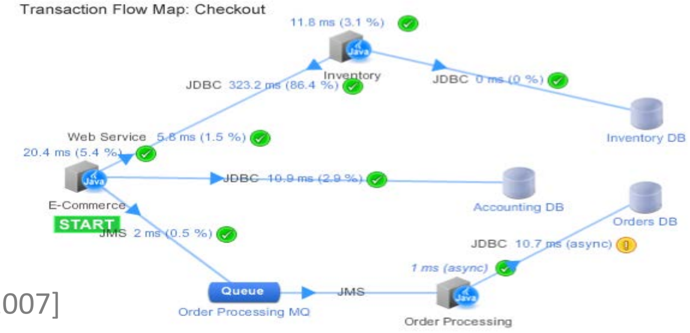
[Panas et al. 2003]

[Wettel and Lanza 2007]

[Greevy et al. 2006]

[Alam and Dugerdil 2007]

APM Tools



[www.appdynamics.com]

Monitoring

Kieker

[van Hoorn et al. 2012]

SPASS-meter

[Eichelberger and Schmid 2014]

Publications & Theses

- 20 publications in 3.5 years
  (15 peer-reviewed)

Selected Publications

- **ICPC 2015**: Fittkau, Finke, Hasselbring, and Waller, *“Comparing Trace Visualizations for Program Comprehension through Controlled Experiments” (Acceptance Rate: 31,5%)*
- **ECIS 2015**: Fittkau, Roth, and Hasselbring, *ExplorViz: Visual Runtime Behavior Analysis of Enterprise Application Landscapes (Acceptance Rate: 31%)*
- **VISSOFT 2013**: Fittkau, Waller, Wulf, and Hasselbring, *Live Trace Visualization for Comprehending Large Software Landscapes: The ExplorViz Approach*
- **ICSE 2013:** Frey, Fittkau, and Hasselbring, *Search-Based Genetic Optimization for Deployment and Reconfiguration of Software in the Cloud*

15 Student‘s Theses in the Context of ExplorViz

[Barbie 2014], [Barzel 2014], [Beye 2013], [Finke 2014], [Gill 2015], [Koppenhagen 2013], [Kosche 2013], [Krause 2015], [Mannstedt 2015], [Matthiessen 2014], [Michaelis 2015], [Simolka 2015], [Stelzer 2014], [Weißenfels 2014], [Zirkelbach 2015]

Outlook & Future Work

Live trace visualization for large software landscapes available as **open-source software** (Apache License 2.0)

ExplorViz

http://www.explorviz.net

All **results available online**

- Raw results, R scripts, code, ratings, …
- ExplorViz versions used in the experiments
- All screen and camera recordings about 160 hours material
- Long-time archival on Zenodo.org

Future Work:

- More controlled experiments (e.g., comparison with more visualization metaphors; professionals as subjects)
- Layout of the Application-Level Perspective [Barbie 2014]

Bibliography

[Alam and Dugerdil 2007] S. Alam and P. Dugerdil, Evospaces: 3D visualization of software architecture. In: *Proc. of 19th Int. Conf. on Soft. Eng. and Know. Eng.* IEEE, 2007, pp. 500–505.

[Barbie 2014] A. Barbie. Stable 3D City Layout in ExplorViz, Bachelor thesis, Kiel University, Kiel, Germany, 93 pp

[Barzel 2014] M. Barzel. Evaluation von Clustering-Verfahren von Klassen für hierarchische Visualisierung in ExplorViz, Bachelor thesis, Kiel University, Kiel, Germany, 57 pp

[Beye 2013] J. Beye. Technology Evaluation for the Communication between the Monitoring and Analysis Component in Kieker, Bachelor thesis, Kiel University, Kiel, Germany, 53 pp

[Eichelberger and Schmid 2014] H. Eichelberger and K. Schmid. Flexible resource monitoring of Java programs. Journal of Systems and Software 93 (July 2014), pages 163–186.

[Finke 2014] S. Finke. Automatische Anleitung einer Versuchsperson während eines kontrollierten Experiments in ExplorViz, Master thesis, Kiel University, Kiel, Germany, 87 pp

[Fittkau et al. 2013a] F. Fittkau, J. Waller, C. Wulf, and W. Hasselbring. Live trace visualization for comprehending large software landscapes: The ExplorViz approach. In: 1st IEEE International Working Conference on Software Visualization (VISSOFT 2013). IEEE Computer Society, Sept. 2013, pages 1–4.

[Fittkau et al. 2013b] F. Fittkau, J. Waller, P. C. Brauer, and W. Hasselbring. Scalable and live trace processing with Kieker utilizing cloud computing. In: Proceedings of the Symposium on Software Performance: Joint Kieker/Palladio Days (KPDays 2013), CEUR, Nov. 2013, pages 89–98.

[Fittkau 2013] F. Fittkau. Live Trace Visualization for System and Program Comprehension in Large Software Landscapes, TR 1310. Department of Computer Science, Kiel University, Germany.

[Fittkau et al. 2014a] F. Fittkau, A. van Hoorn, and W. Hasselbring. Towards a dependability control center for large software landscapes. In: Tenth European Dependable Computing Conference (EDCC 2014), May 2014.

[Fittkau et al. 2014b] F. Fittkau, P. Stelzer, and W. Hasselbring. Live visualization of large software landscapes for ensuring architecture conformance. In: ECSAW 2nd International Workshop on Software Engineering for Systems-of-Systems (SESoS 2014), Aug. 2014.

[Fittkau et al. 2015a] Fittkau, F., Finke, S., Hasselbring, W. and Waller, J. Comparing Trace Visualizations for Program Comprehension through Controlled Experiments. In: 23rd IEEE International Conference on Program Comprehension (ICPC 2015), May 2015

[Fittkau et al. 2015b] Fittkau, F., Roth, S. and Hasselbring, W. ExplorViz: Visual Runtime Behavior Analysis of Enterprise Application Landscapes. In: 23rd European Conference on Information Systems (ECIS 2015), May 2015

[Gill 2015] J. Gill. Integration von Kapazitätsmanagement in ein Kontrollzentrum für Softwarelandschaften, Bachelor thesis, Kiel University, Kiel, Germany, 49 pp

[Greevy et al. 2006] O. Greevy, M. Lanza, and C. Wysseier, Visualizing live software systems in 3D. In: *Proc. of the 2006 ACM Symposium on Software Visualization*, 2006, pp. 47–56.

[Jähde 2015] D. Jähde, Performance Analyse Tool in ExplorViz, WiSe 2015. Department of Computer Science, Kiel University, Kiel, Germany.

[Koppenhagen 2013] E. Koppenhagen. Evaluation von Elastizitätsstrategien in der Cloud im Hinblick auf optimale Ressourcennutzung ,Bachelor thesis, Kiel University, Kiel, Germany, 67 pp

[Kosche 2013] M. Kosche. Tracking User Actions for the Web-Based Front End of ExplorViz, Bachelor thesis, Kiel University, Kiel, Germany, 51 pp

[Matthiessen 2014] N. Matthiessen. Monitoring Remote Procedure Calls - Concepts and Evaluation, Bachelor thesis, Kiel University, Kiel, Germany, 43 pp

[Mannstedt 2015] K. C. Mannstedt. Integration von Anomalienerkennung in einem Kontrollzentrum für Softwarelandschaften, Bachelor thesis, Kiel University, Kiel, Germany, 53 pp

[Michaelis 2015] J. Michaelis. Integration von Ursachenerkennung in ein Kontrollzentrum für Softwarelandschaften, Bachelor thesis, Kiel University, Kiel, Germany, 48 pp

[Panas et al. 2003] T. Panas, R. Berrigan, and J. Grundy, A 3D metaphor for software production visualization. In: *Proceedings of the 7th International Conference on Information Visualization (IV 2003)*. IEEE Computer Society, 2003, pp. 314–320.

Bibliography (cont'd)

[Stelzer 2014] P. Stelzer. Scalable and Live Trace Processing in the Cloud, Bachelor thesis, Kiel University, Kiel, Germany, 52 pp.

[van Hoorn et al. 2012] A. van Hoorn, J. Waller, and W. Hasselbring. Kieker: A framework for application performance monitoring and dynamic software analysis. In: Proceedings of the 3rd ACM/SPEC International Conference on PerformanceEngineering (ICPE 2012). ACM, Apr. 2012, pages 247–248.

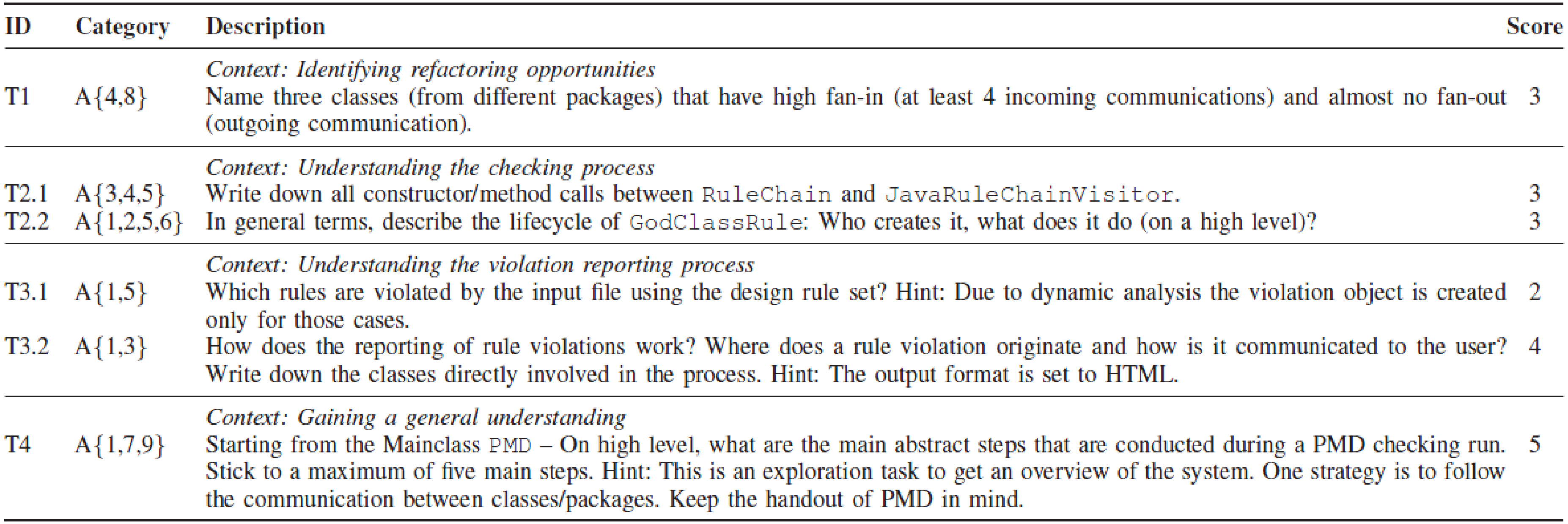
[Waller et al. 2014] J. Waller, F. Fittkau, and W. Hasselbring. Application performance monitoring: Trade-off between overhead reduction and maintainability. In: Proceedings of the Symposium on Software Performance: Joint Descartes/Kieker/Palladio Days (SoSP 2014). Nov. 2014, pages 1–24.

[Weißenfels 2014] B. Weißenfels. Evaluation of Trace Reduction Techniques for Online Trace Visualization, Master thesis, Kiel University, Kiel, Germany, 82 pp

[Wettel and Lanza 2007] R. Wettel and M. Lanza. Visualizing software systems as cities. In: *Proc. 4th Int. Workshop on Visualizing Software for Understanding and Analysis (VISSOFT 2007)*, 2007.

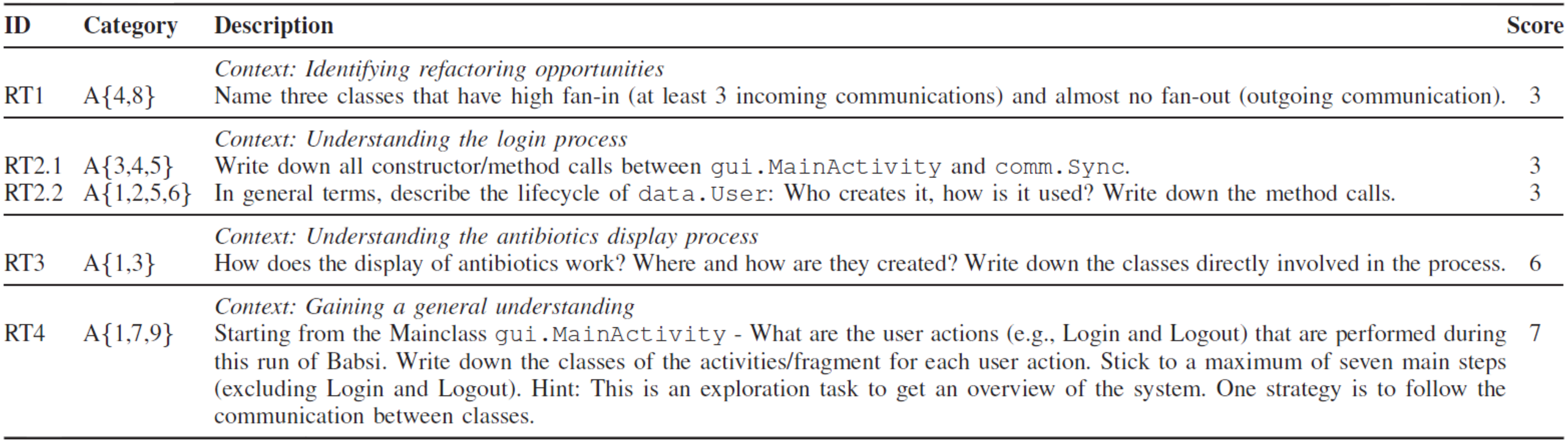
Backup Slides

Tasks Application Experiment



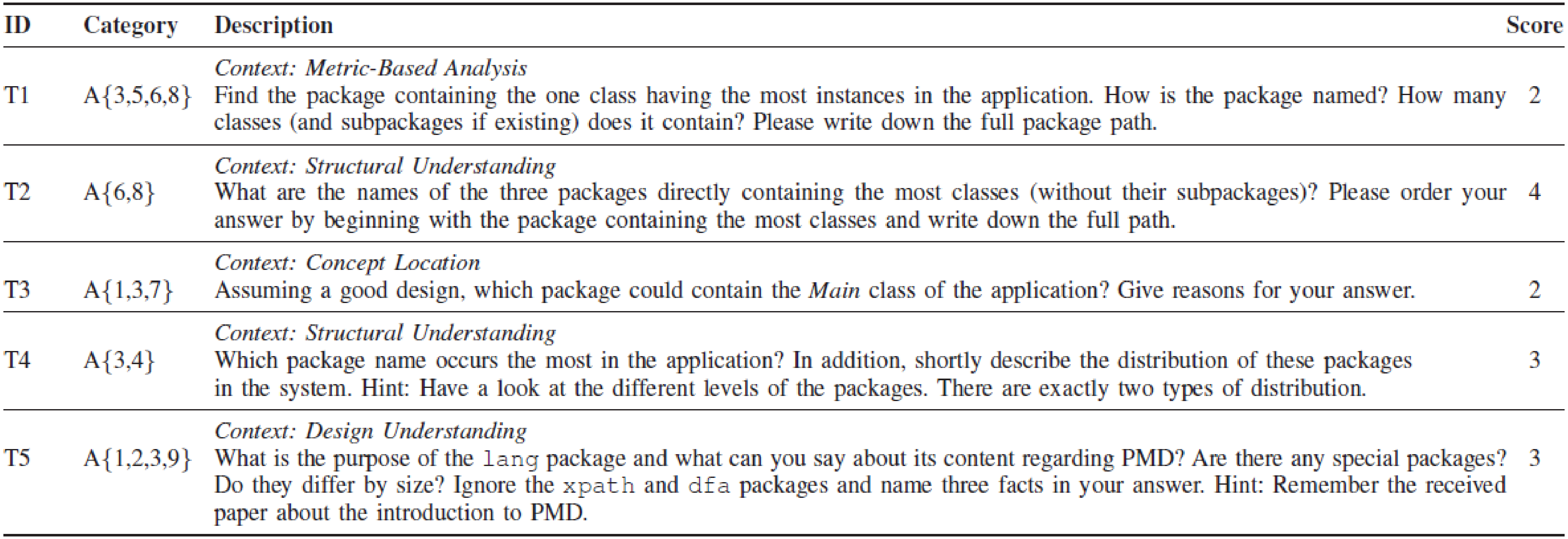
ID	Category	Description	Score
		Context: Identifying refactoring opportunities	
T1	A{4,8}	Name three classes (from different packages) that have high fan-in (at least 4 incoming communications) and almost no fan-out (outgoing communication).	3
		Context: Understanding the checking process	
T2.1	A{3,4,5}	Write down all constructor/method calls between `RuleChain` and `JavaRuleChainVisitor`.	3
T2.2	A{1,2,5,6}	In general terms, describe the lifecycle of `GodClassRule`: Who creates it, what does it do (on a high level)?	3
		Context: Understanding the violation reporting process	
T3.1	A{1,5}	Which rules are violated by the input file using the design rule set? Hint: Due to dynamic analysis the violation object is created only for those cases.	2
T3.2	A{1,3}	How does the reporting of rule violations work? Where does a rule violation originate and how is it communicated to the user? Write down the classes directly involved in the process. Hint: The output format is set to HTML.	4
		Context: Gaining a general understanding	
T4	A{1,7,9}	Starting from the Mainclass `PMD` – On high level, what are the main abstract steps that are conducted during a PMD checking run. Stick to a maximum of five main steps. Hint: This is an exploration task to get an overview of the system. One strategy is to follow the communication between classes/packages. Keep the handout of PMD in mind.	5

Tasks Application Experiment 2



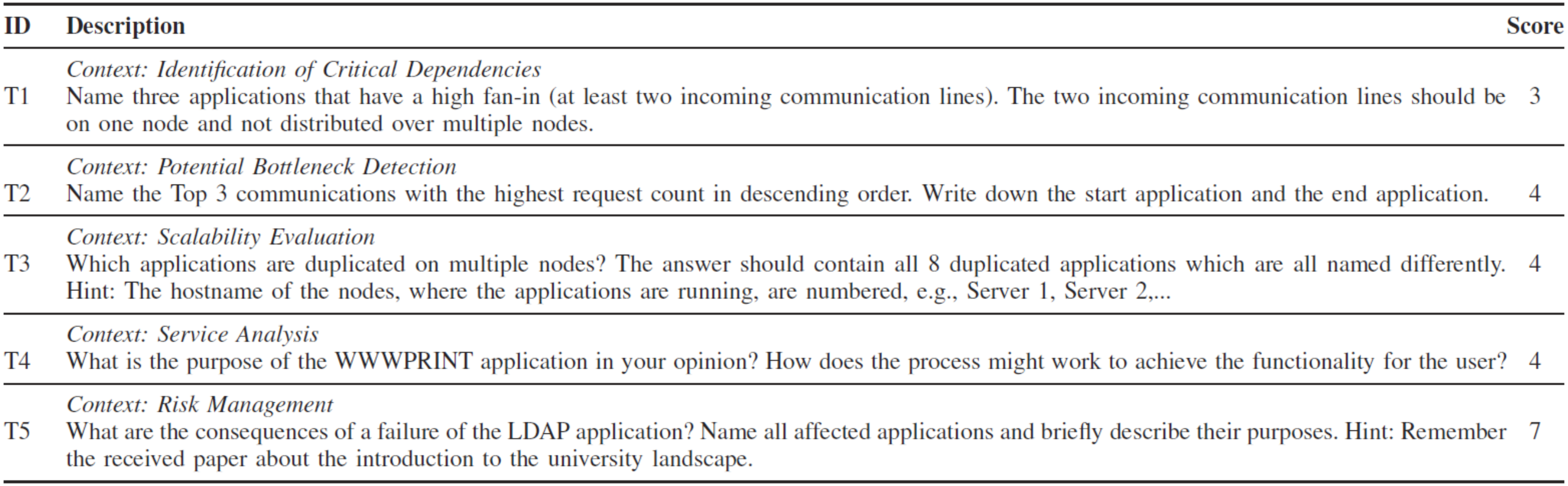
ID	Category	Description	Score
		Context: Identifying refactoring opportunities	
RT1	A{4,8}	Name three classes that have high fan-in (at least 3 incoming communications) and almost no fan-out (outgoing communication).	3
		Context: Understanding the login process	
RT2.1	A{3,4,5}	Write down all constructor/method calls between `gui.MainActivity` and `comm.Sync`.	3
RT2.2	A{1,2,5,6}	In general terms, describe the lifecycle of `data.User`: Who creates it, how is it used? Write down the method calls.	3
		Context: Understanding the antibiotics display process	
RT3	A{1,3}	How does the display of antibiotics work? Where and how are they created? Write down the classes directly involved in the process.	6
		Context: Gaining a general understanding	
RT4	A{1,7,9}	Starting from the Mainclass `gui.MainActivity` - What are the user actions (e.g., Login and Logout) that are performed during this run of Babsi. Write down the classes of the activities/fragment for each user action. Stick to a maximum of seven main steps (excluding Login and Logout). Hint: This is an exploration task to get an overview of the system. One strategy is to follow the communication between classes.	7

Tasks 3D Print Experiment



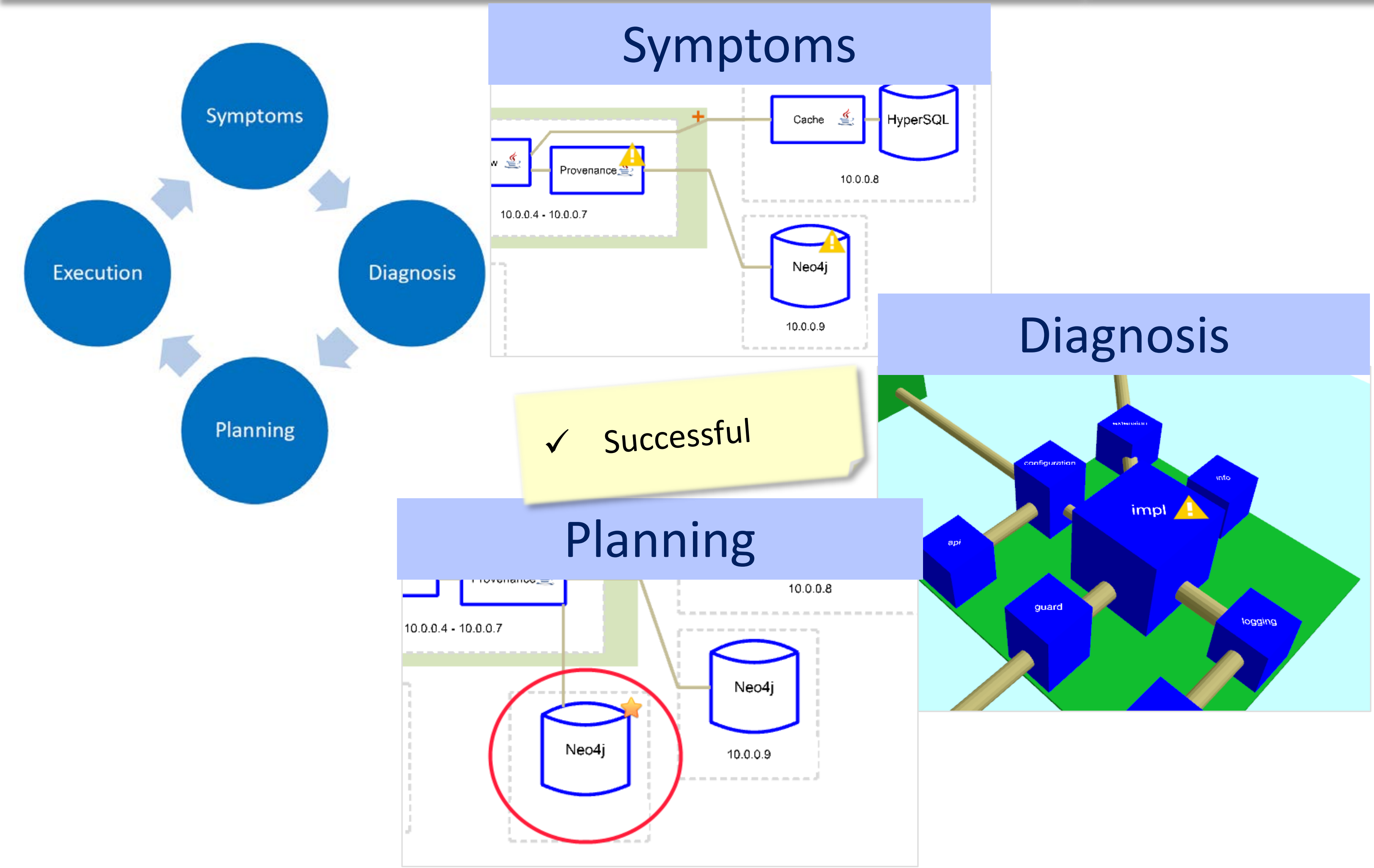
ID	Category	Description	Score
T1	A{3,5,6,8}	*Context: Metric-Based Analysis* Find the package containing the one class having the most instances in the application. How is the package named? How many classes (and subpackages if existing) does it contain? Please write down the full package path.	2
T2	A{6,8}	*Context: Structural Understanding* What are the names of the three packages directly containing the most classes (without their subpackages)? Please order your answer by beginning with the package containing the most classes and write down the full path.	4
T3	A{1,3,7}	*Context: Concept Location* Assuming a good design, which package could contain the *Main* class of the application? Give reasons for your answer.	2
T4	A{3,4}	*Context: Structural Understanding* Which package name occurs the most in the application? In addition, shortly describe the distribution of these packages in the system. Hint: Have a look at the different levels of the packages. There are exactly two types of distribution.	3
T5	A{1,2,3,9}	*Context: Design Understanding* What is the purpose of the `lang` package and what can you say about its content regarding PMD? Are there any special packages? Do they differ by size? Ignore the `xpath` and `dfa` packages and name three facts in your answer. Hint: Remember the received paper about the introduction to PMD.	3

Tasks Landscape Experiment



ID	Description	Score
T1	*Context: Identification of Critical Dependencies* Name three applications that have a high fan-in (at least two incoming communication lines). The two incoming communication lines should be on one node and not distributed over multiple nodes.	3
T2	*Context: Potential Bottleneck Detection* Name the Top 3 communications with the highest request count in descending order. Write down the start application and the end application.	4
T3	*Context: Scalability Evaluation* Which applications are duplicated on multiple nodes? The answer should contain all 8 duplicated applications which are all named differently. Hint: The hostname of the nodes, where the applications are running, are numbered, e.g., Server 1, Server 2,...	4
T4	*Context: Service Analysis* What is the purpose of the WWWPRINT application in your opinion? How does the process might work to achieve the functionality for the user?	4
T5	*Context: Risk Management* What are the consequences of a failure of the LDAP application? Name all affected applications and briefly describe their purposes. Hint: Remember the received paper about the introduction to the university landscape.	7

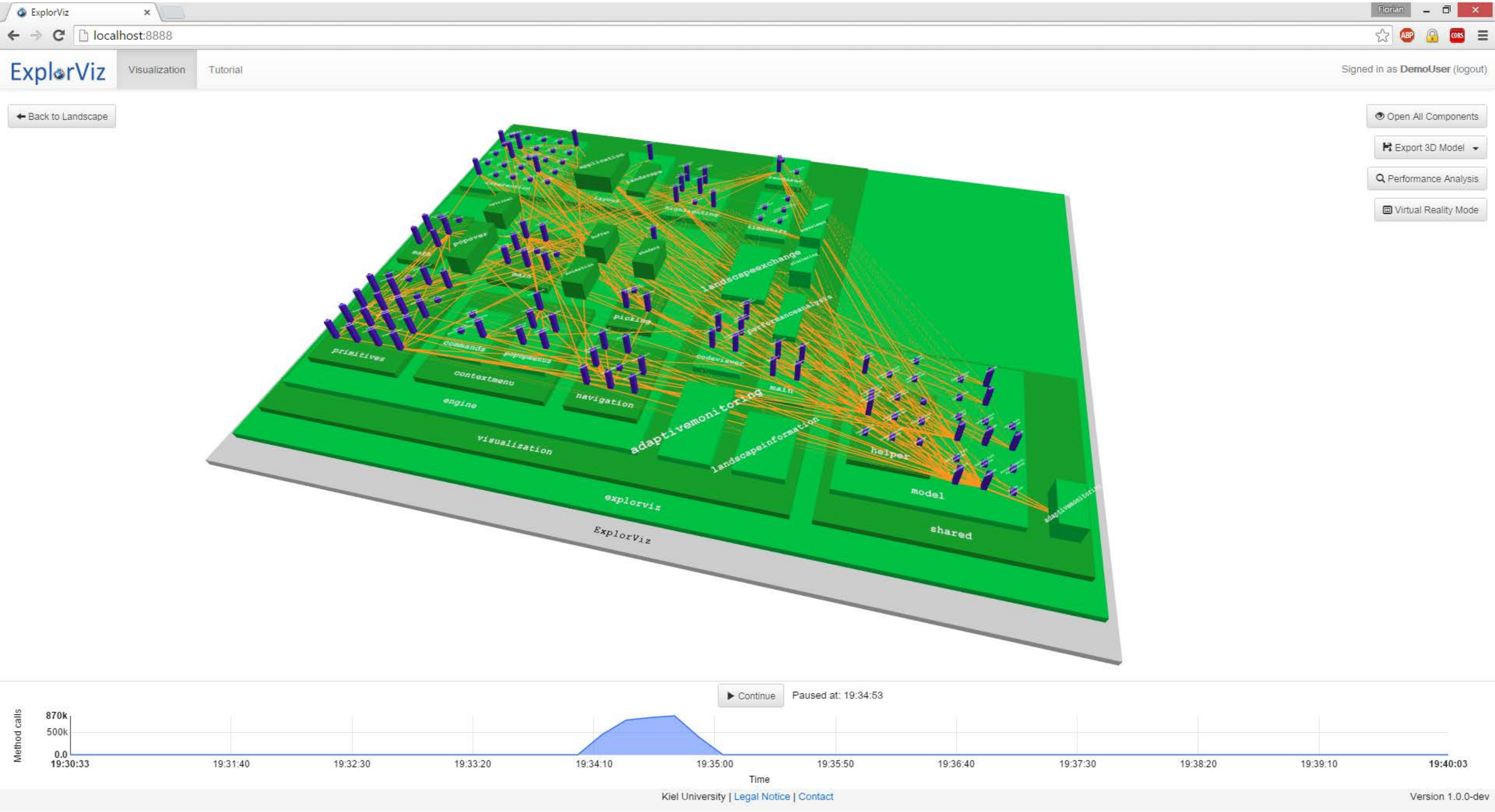
Extensibility (Control Center)



✓ Successful

[Fittkau et al. 2014a, Gill 2015, Michaelis 2015, Mannstedt 2015]

Meta-Monitoring


ExplorViz
localhost:8888
Florian
ExplorViz
Visualization
Tutorial
Signed in as DemoUser (logout)
Back to Landscape
Open All Components
Export 3D Model
Performance Analysis
Virtual Reality Mode
application
landscape
interaction
layout
highlighting
timeshift
popover
main
picking
landscapeexchange
performanceanalysis
primitives
commands
popupmenus
contextmenu
navigation
codeviewer
engine
main
visualization
adaptivemonitoring
landscapeinformation
helper
model
explorviz
shared
ExplorViz
adaptivemonitoring
Continue
Paused at: 19:34:53
Method calls
870k
500k
0.0
19:30:33
19:31:40
19:32:30
19:33:20
19:34:10
19:35:00
19:35:50
19:36:40
19:37:30
19:38:20
19:39:10
19:40:03
Time
Kiel University | Legal Notice | Contact
Version 1.0.0-dev