

Posrednici umreženih sustava

Prof. dr. sc. Siniša Srbljić

Prof. dr. sc. Dalibor Vrsalović

Dr. sc. Ivan Skuliber

Fakultet elektrotehnike i računarstva,
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sisteme



Quality Impact PRediction for Evolving Software Systems (Q-ImPrESS)

Dr.sc. Ivan Skuliber

Outline

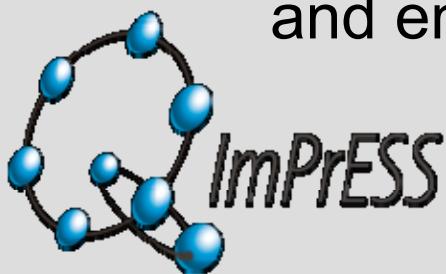
- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Outline

- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Q-ImPrESS FP7 project

- **Quality Impact PRediction for Evolving Software Systems**
 - 7th Framework Programme
 - Started at the beginning of 2008
 - Finished at the end of 2010
- **Small or medium-scale focused research project**
 - “Strategic objective” (defined by EU)
 - Service and software architectures, infrastructures and engineering



Q-ImPrESS strategic objective

- Service/software engineering approaches and tools for dynamically composed service-oriented systems with dependable quality of service and reliability properties, promoting new open development paradigms with a higher degree of joint user and development community involvement
- Strategies and technologies enabling mastery of complexity, dependability and behavioural stability in complex systems and in systems evolving over time without central design, with guaranteed end-to-end quality of service

Where is the middleware?

- **Implemented software in the project**
 - Uses many different types of middleware
 - Infrastructure, communication, database, ...
- **The use of project results**
 - Predictions of middleware's characteristics
 - Performance, reliability and maintainability

Outline

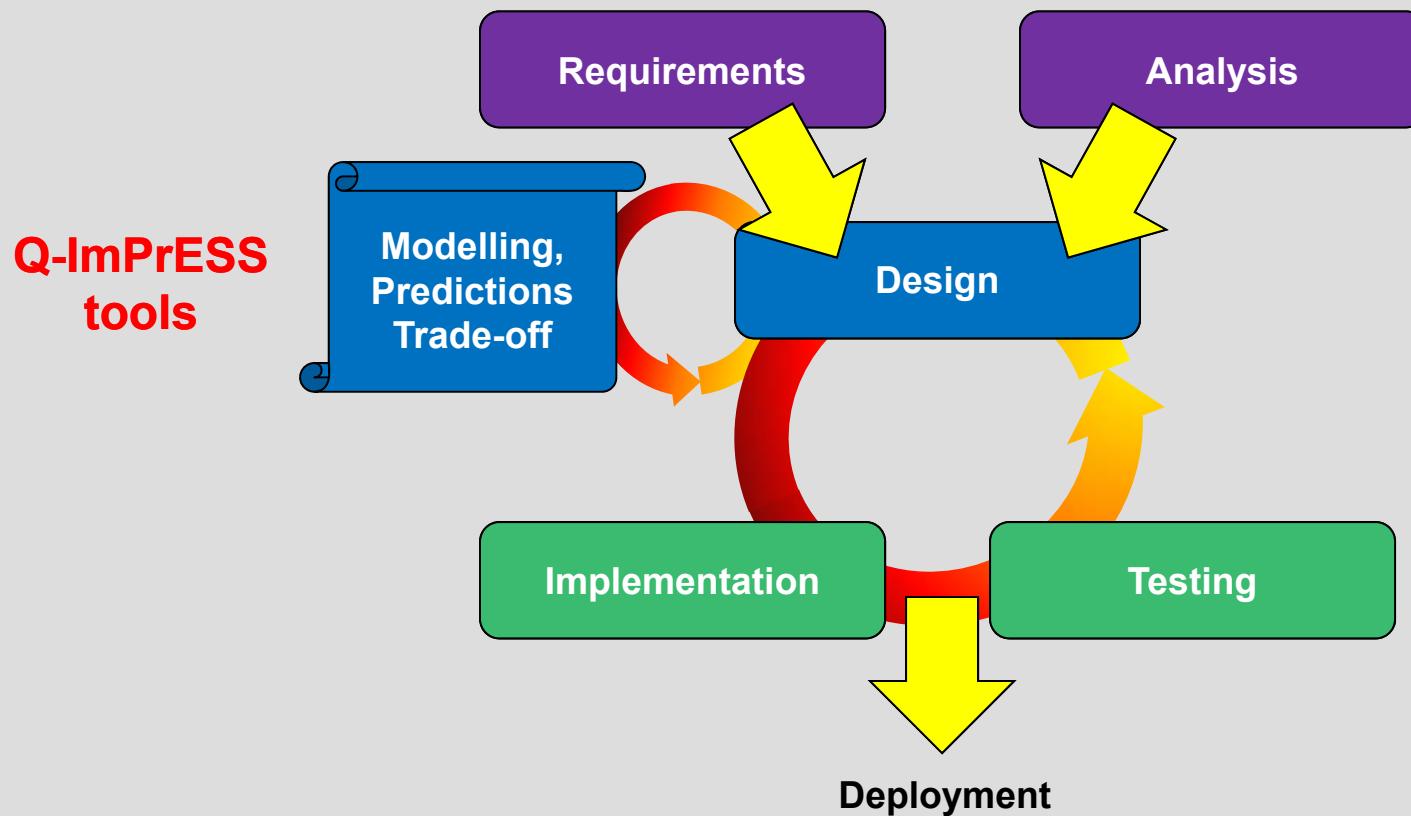
- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Q-ImPrESS consortium

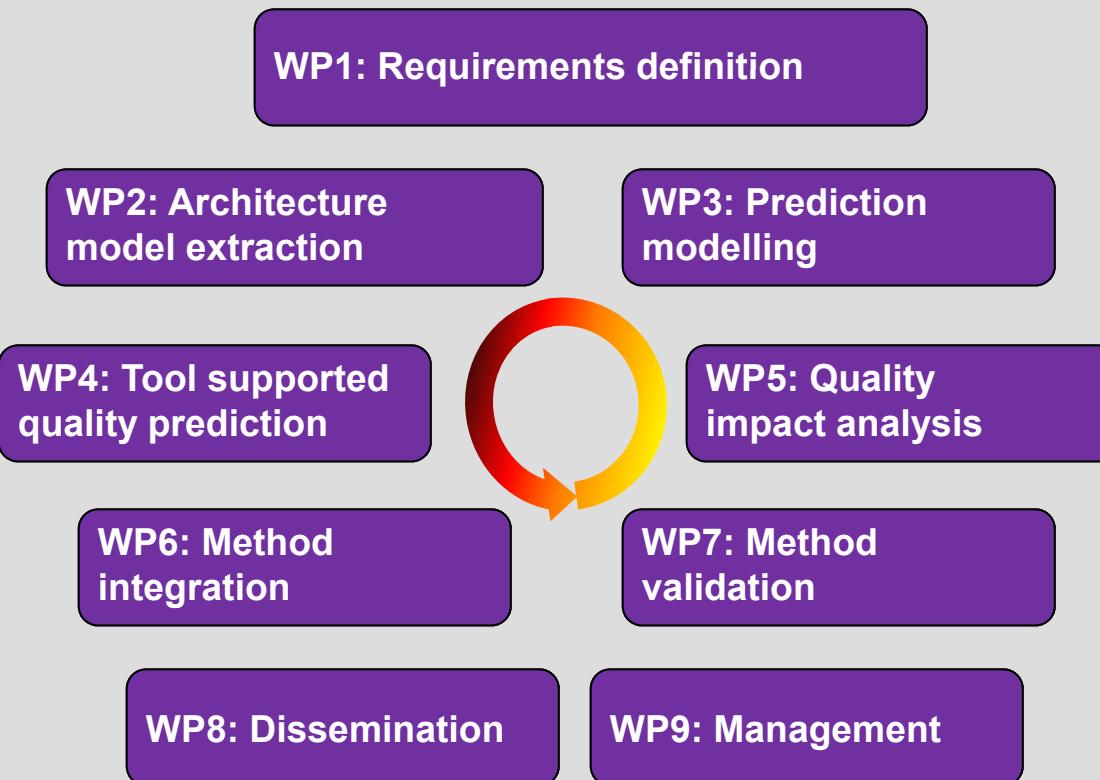


Q-ImPrESS goals

- New software engineering method and tools
- Modelling and monitoring of service-oriented software
- Quality impact prediction, what-if analysis



Work division - workpackages



- Nine workpackages
- Each partner governs one workpackage
- Iterative process
- Our involvement
 - Major: WP1, WP6, WP7
 - Minor: WP8, WP9
 - Leaders of WP1, WP7

Outline

- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Q-ImPrESS usage example

Reliability



0,01%
0,10%

Maintainability

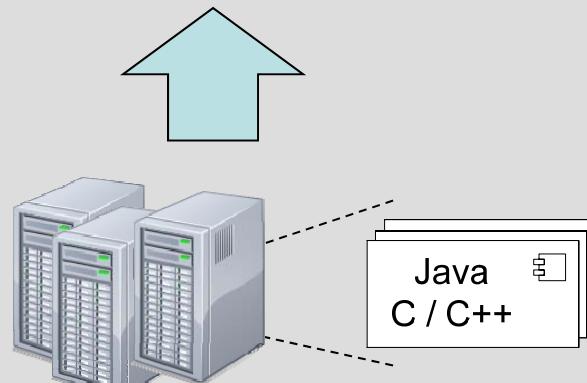
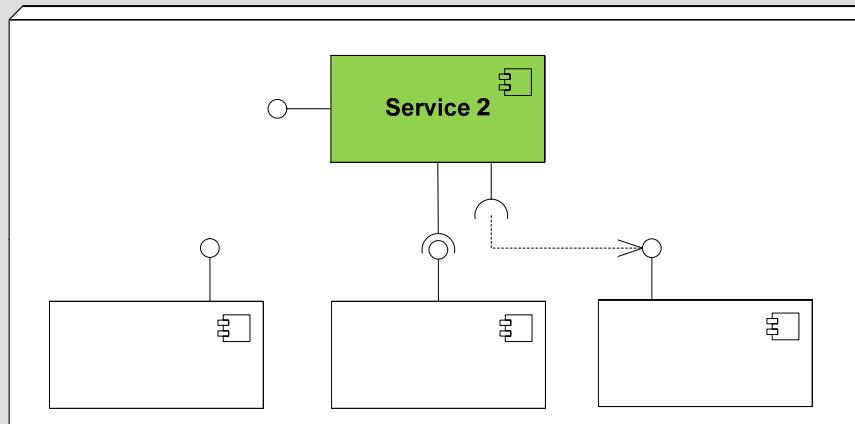


20.000 €
25.000 €

Performance

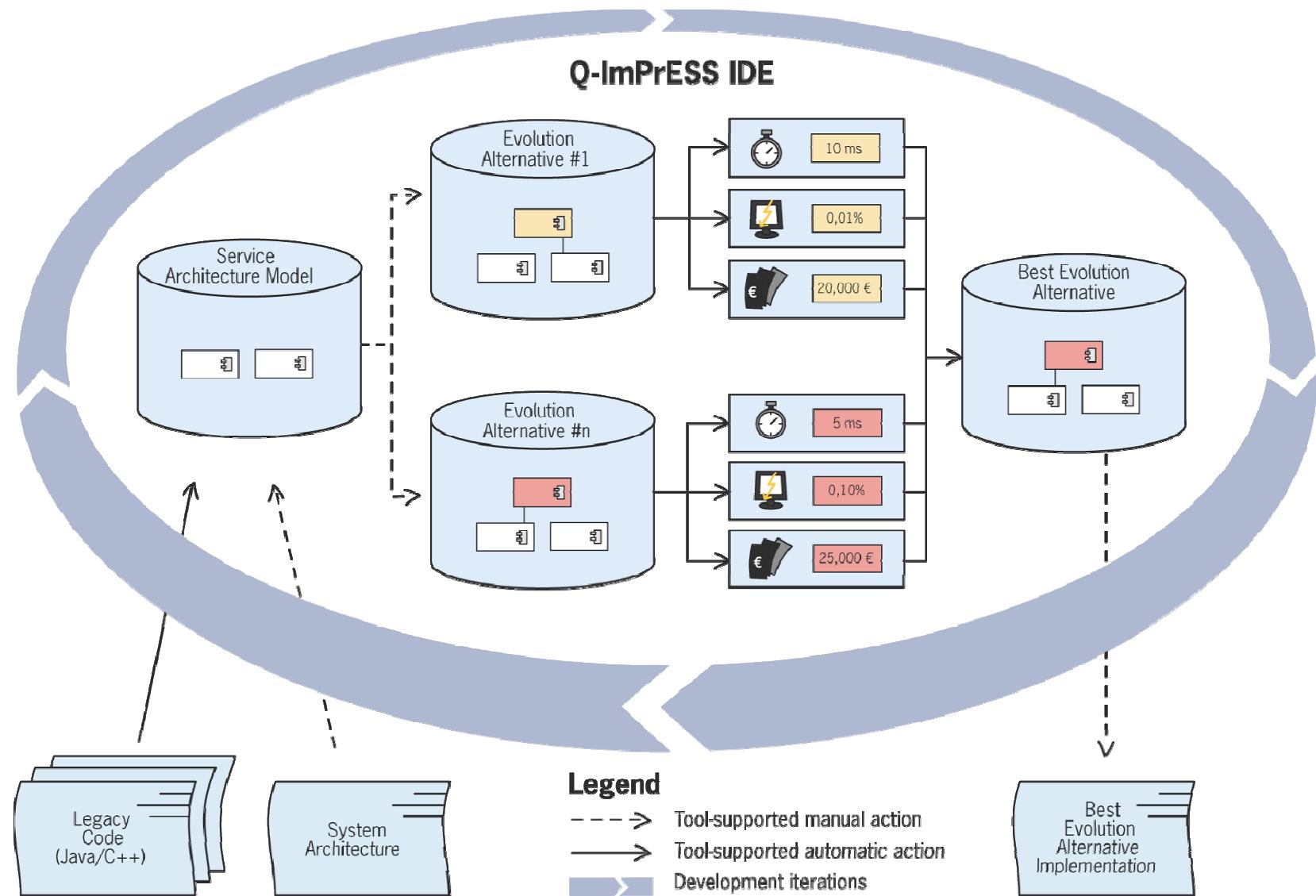


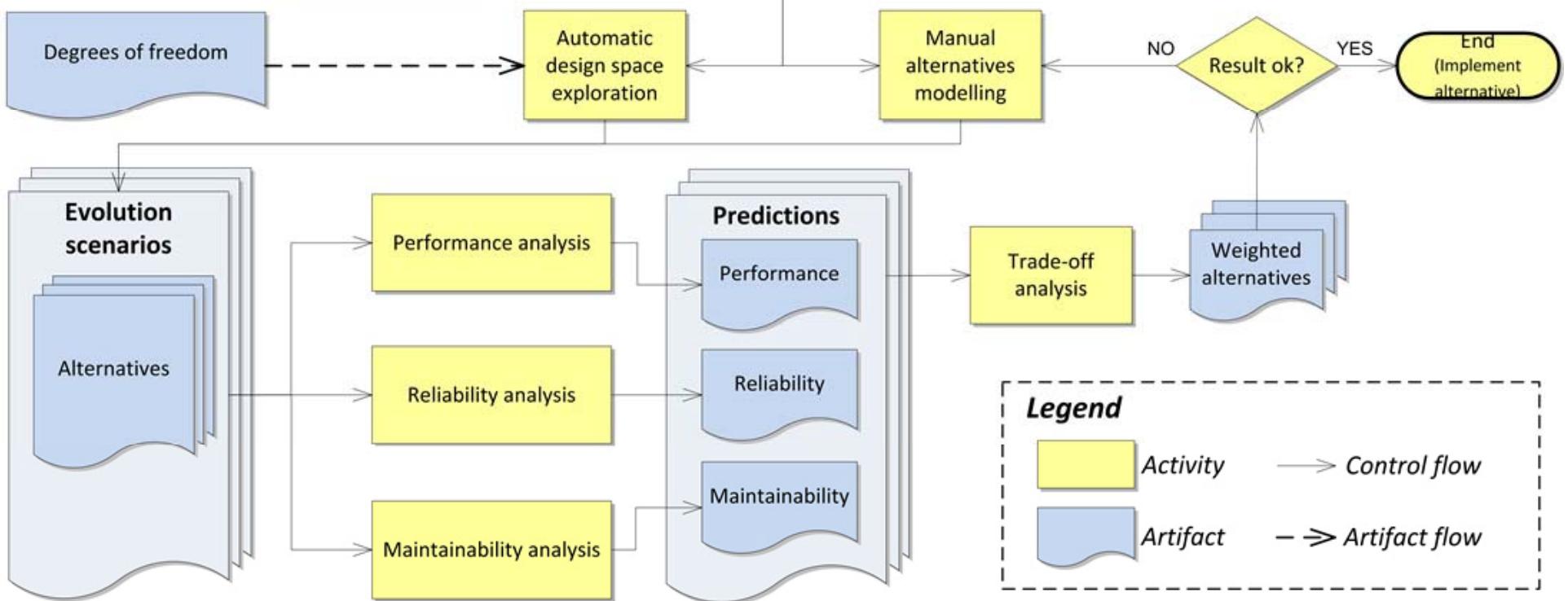
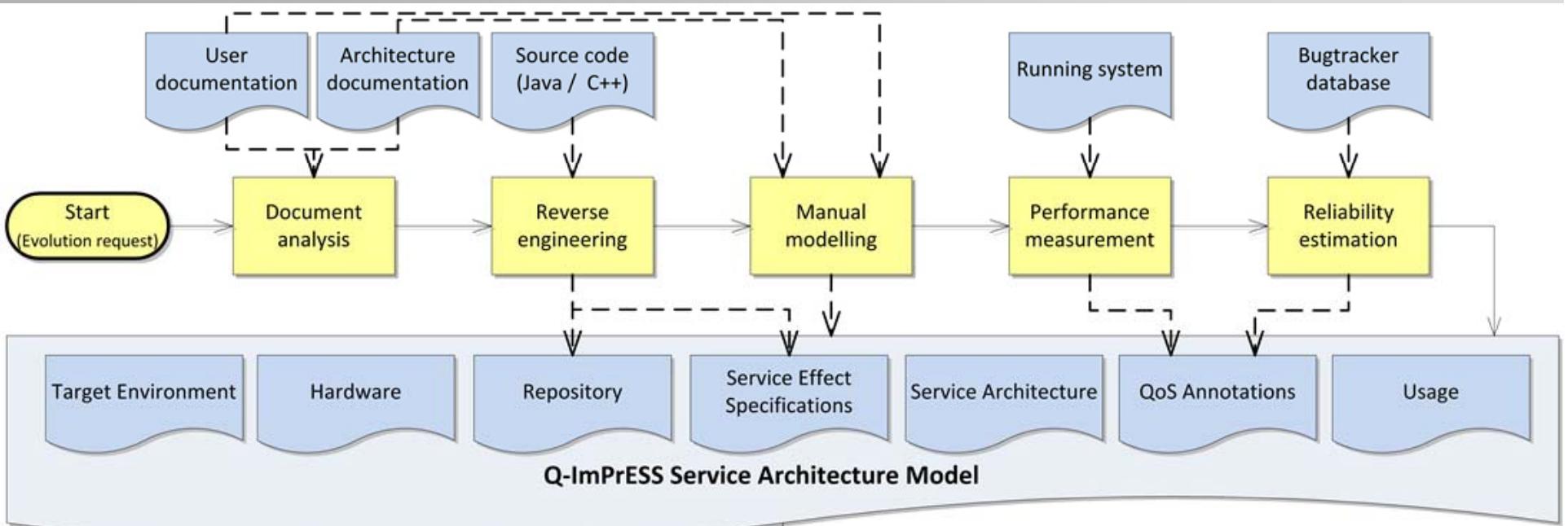
10 ms
5 ms



Legacy Application

Q-ImPrESS usage – process example





Q-ImPrESS usage – process example

Assess alternatives

Please rate actual values to be compared using AHP analysis

Value	EX	VS	ST	MO	EQ	MO	ST	VS	EX	Value
New mass query implementation; Utilization: 14,43	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Main Alternative; Utilization: 8,63					
New mass query implementation; Utilization: 14,43	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Utilization: 49,10
Main Alternative; Utilization: 8,63	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Utilization: 49,10
New mass query implementation; Cost: 6000,0;Time effort: 6,0	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Main Alternative
New mass query implementation; Cost: 6000,0;Time effort: 6,0	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Cost: 3000,0;Time effort: 3,0
Main Alternative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Cost: 3000,0;Time effort: 3,0
New mass query implementation; Reliability: 0,998800	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Main Alternative; Reliability: 0,992237
New mass query implementation; Reliability: 0,998800	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Reliability: 0,974688
Main Alternative; Reliability: 0,992237	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Reliability: 0,974688
New mass query implementation; Response time: 11,97	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Main Alternative; Response time: 21,46
New mass query implementation; Response time: 11,97	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Response time: 16,28
Main Alternative; Response time: 21,46	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	With database cache; Response time: 16,28

Legend of quality ratings:

- EX Extreme
- VS Very Strong
- ST Strong
- MO Moderate
- EQ Equal

Interpret results

AHP Analysis Results

Alternative	Total	Utilization (0,7622)	Cost (0,4853)	Reliability (0,6556)	Response Time (2,0969)
New mass query implementation	6,7322	1,6053	0,2037	0,6429	4,0802
Main Alternative	2,5892	0,5542	0,6352	0,6429	0,3569
With database cache	2,6706	0,1272	0,4170	0,2010	1,0594

Normalize values

Export results as:

Running system → **Performance measurement** → **Reliability estimation**

Bugtracker database → **Performance measurement** → **Reliability estimation**

CompositeComponent serverComposite

```

Repository "876324" : "exampleSystemRepository" {
    doc: "examplesystem Repository Documentation",
    componentType: {
        PrimitiveComponent "25627" : "databasePC" {
            doc: "databasePC documentation",
            required: {},
            provided: {
                InterfacePort "17000" : "databasePC_Provided_DatabaseInterface"
            }
        }
    }
}
  
```

Check Workorganisation Annotation

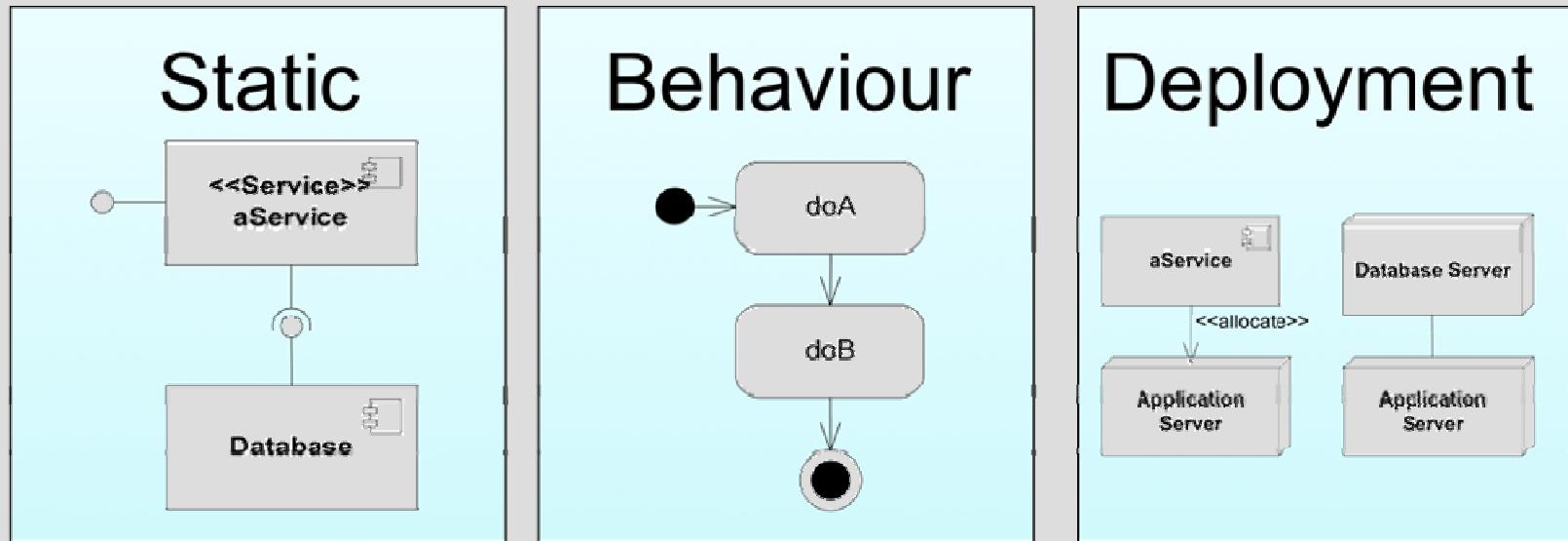
Time

Response Time of scenarioA

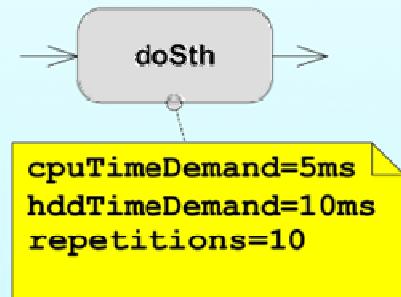
Outline

- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

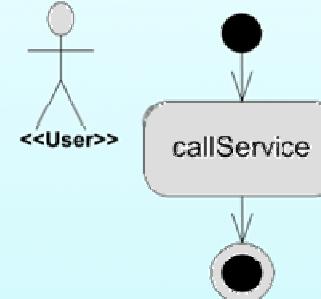
How to model a software system

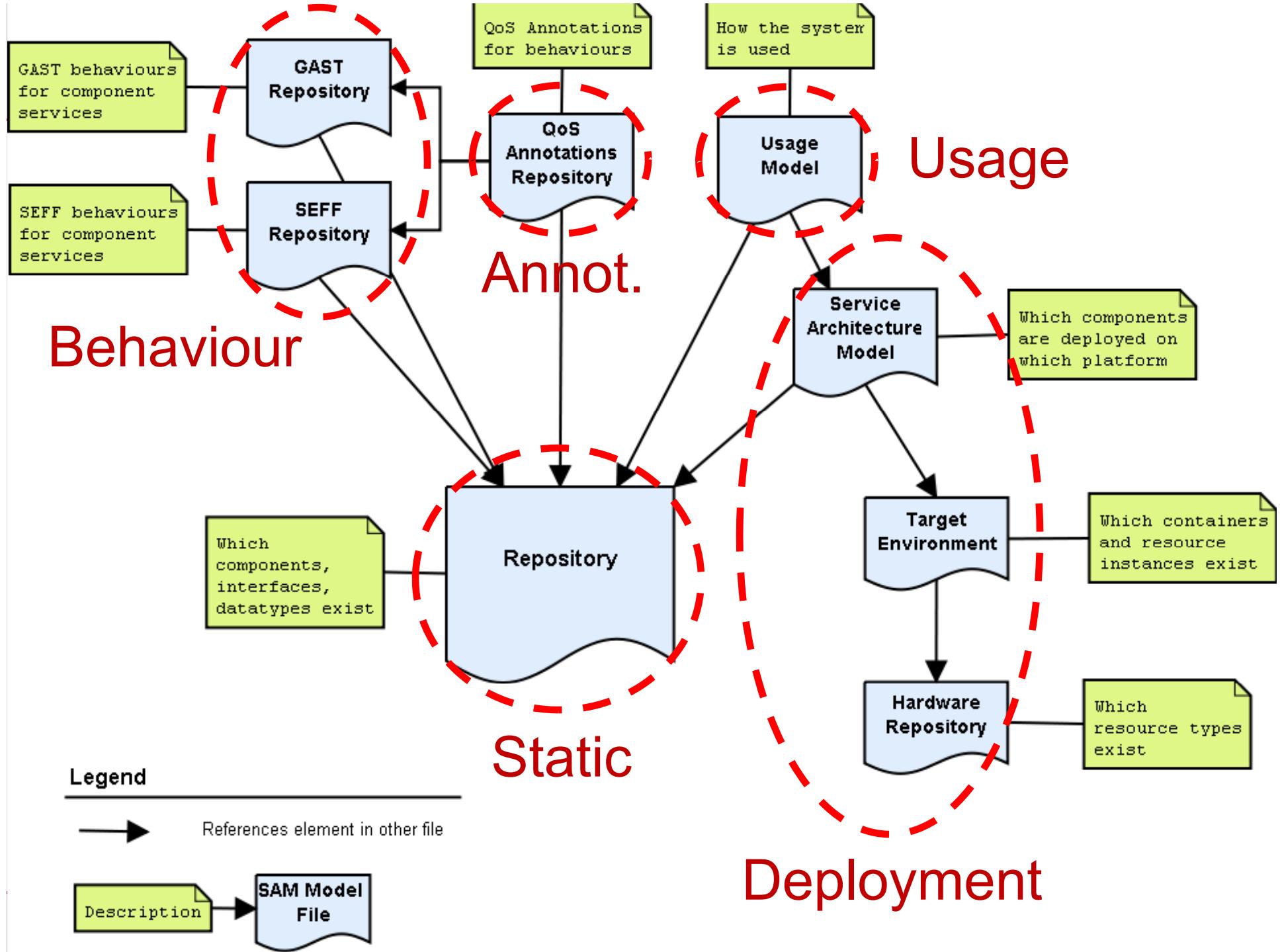


Annotations

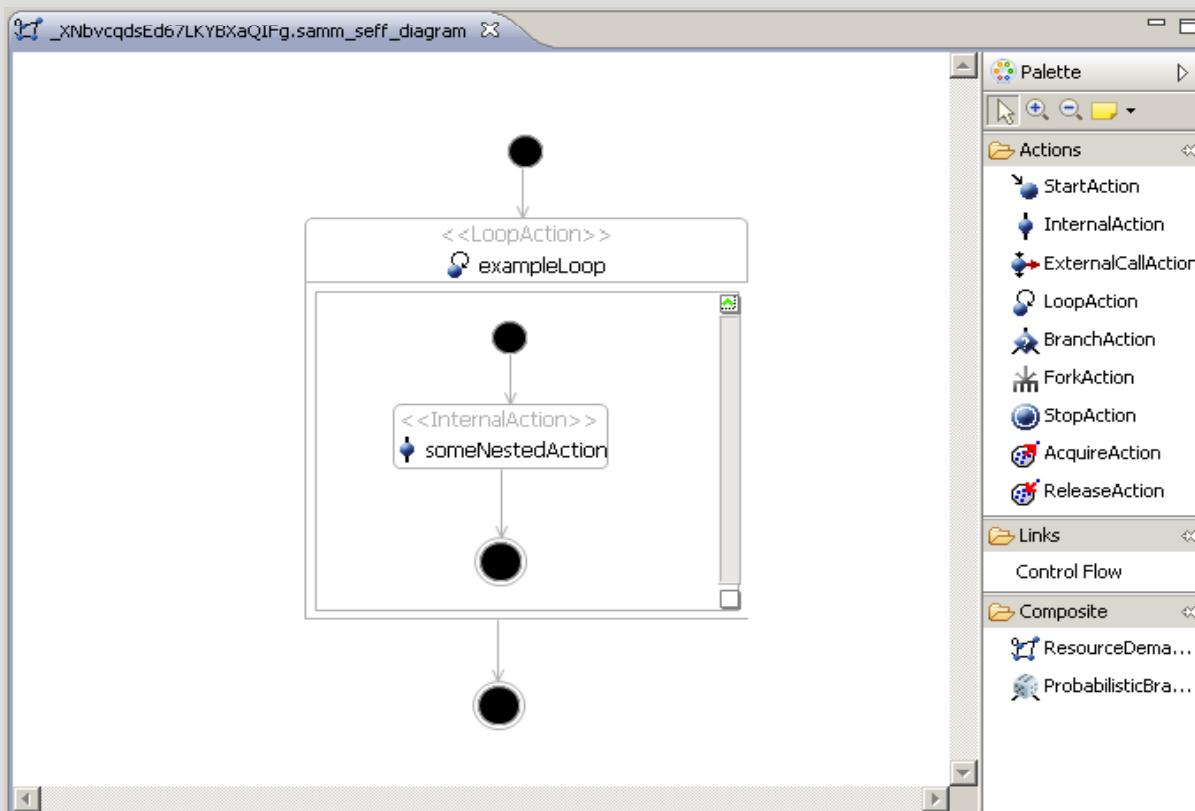


Usage





Behaviour and Annotations model



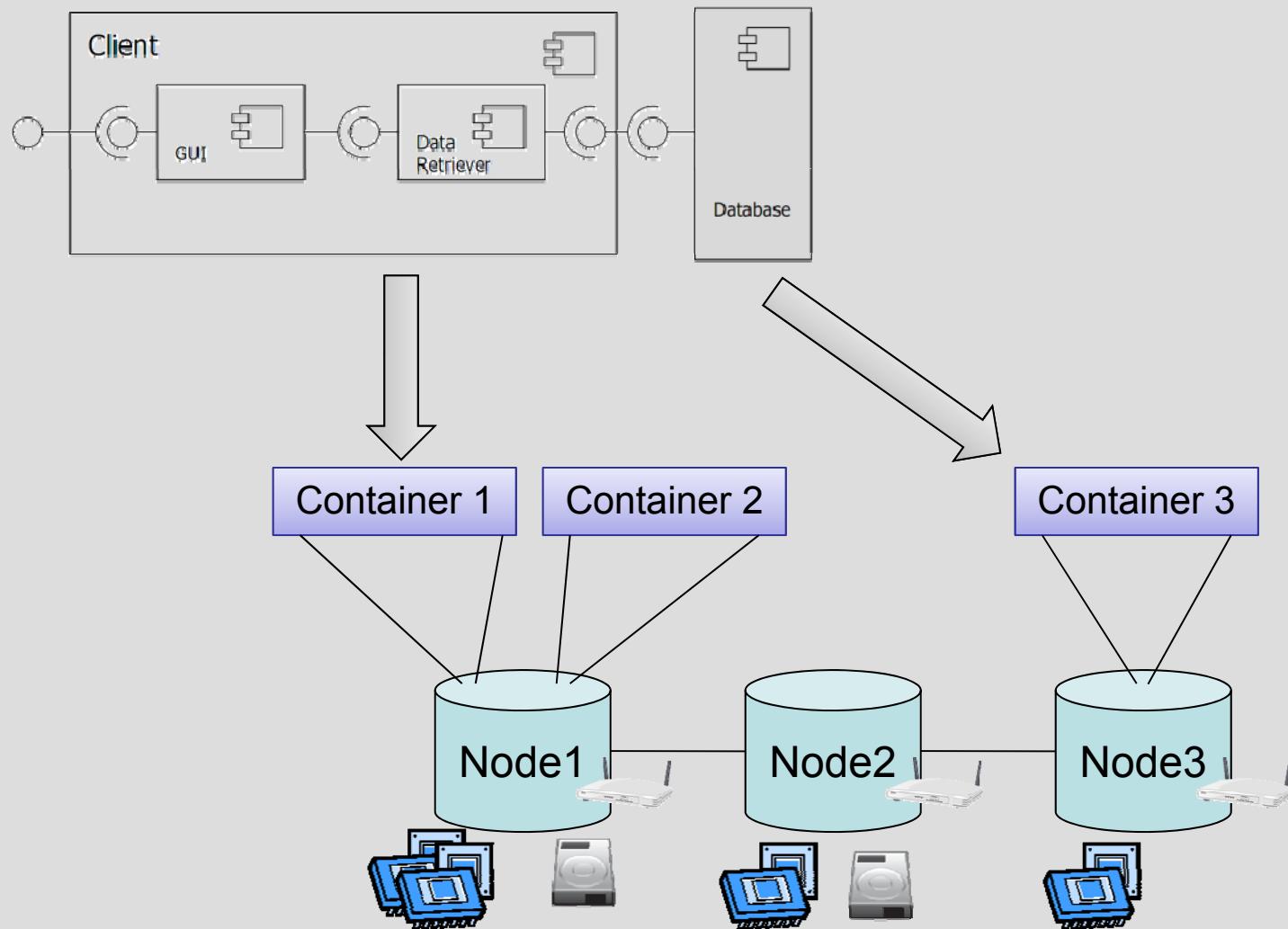
Modelling internals
of a component

Resource use for
every activity in a
component

The properties view shows a resource named "Loop Count aName" with a value of "Constant Number 3.0". The properties table below lists the "Value" property with the value "3.0".

Property	Value
Value	3.0

Deployment model



Outline

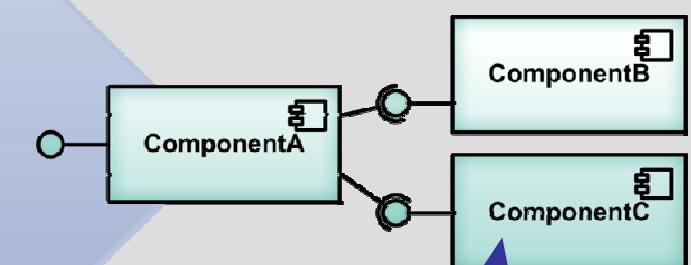
- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Reverse engineering tools



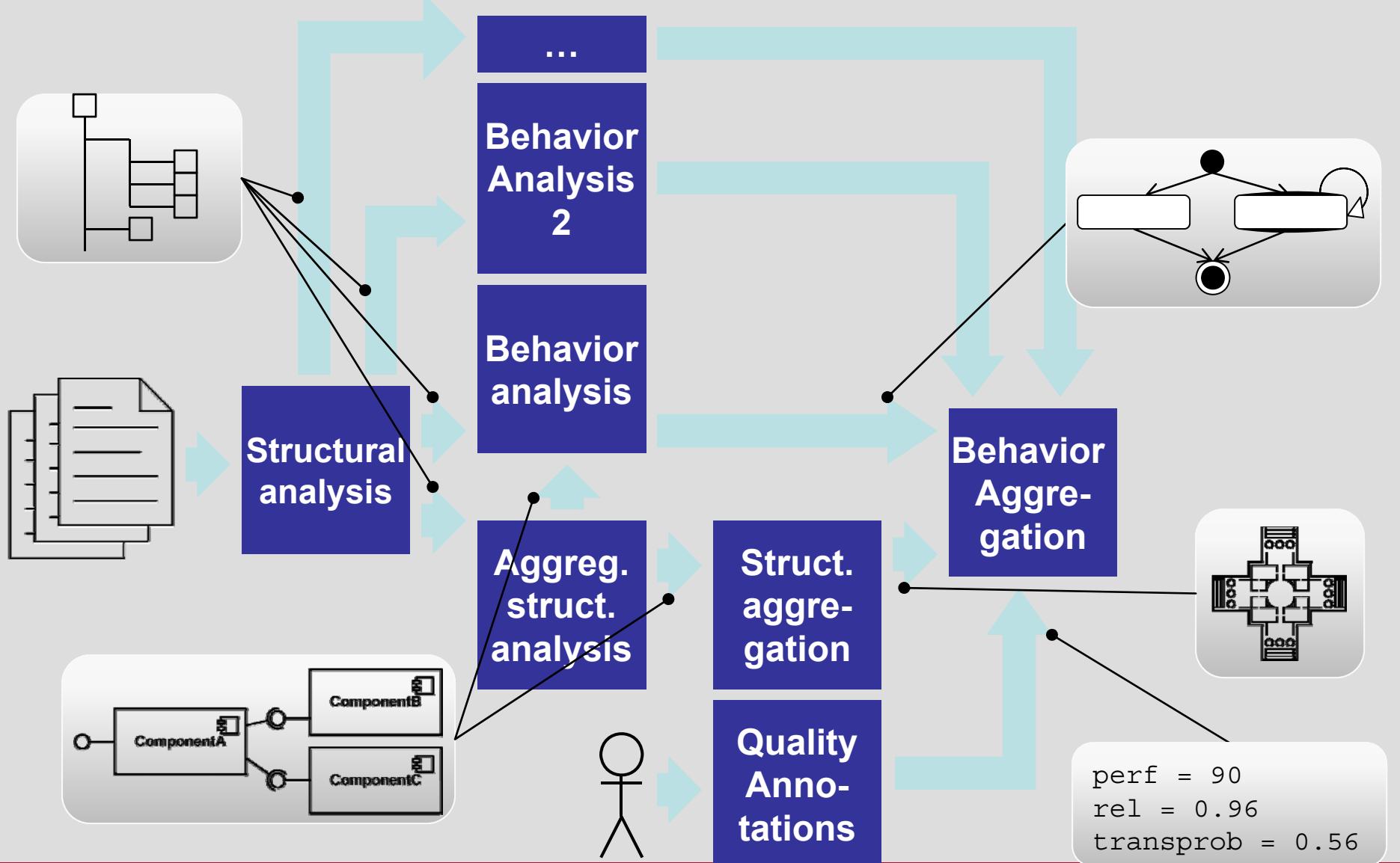
Reverse Engineering

C/C++, Java &
Delphi Code
→ extendable

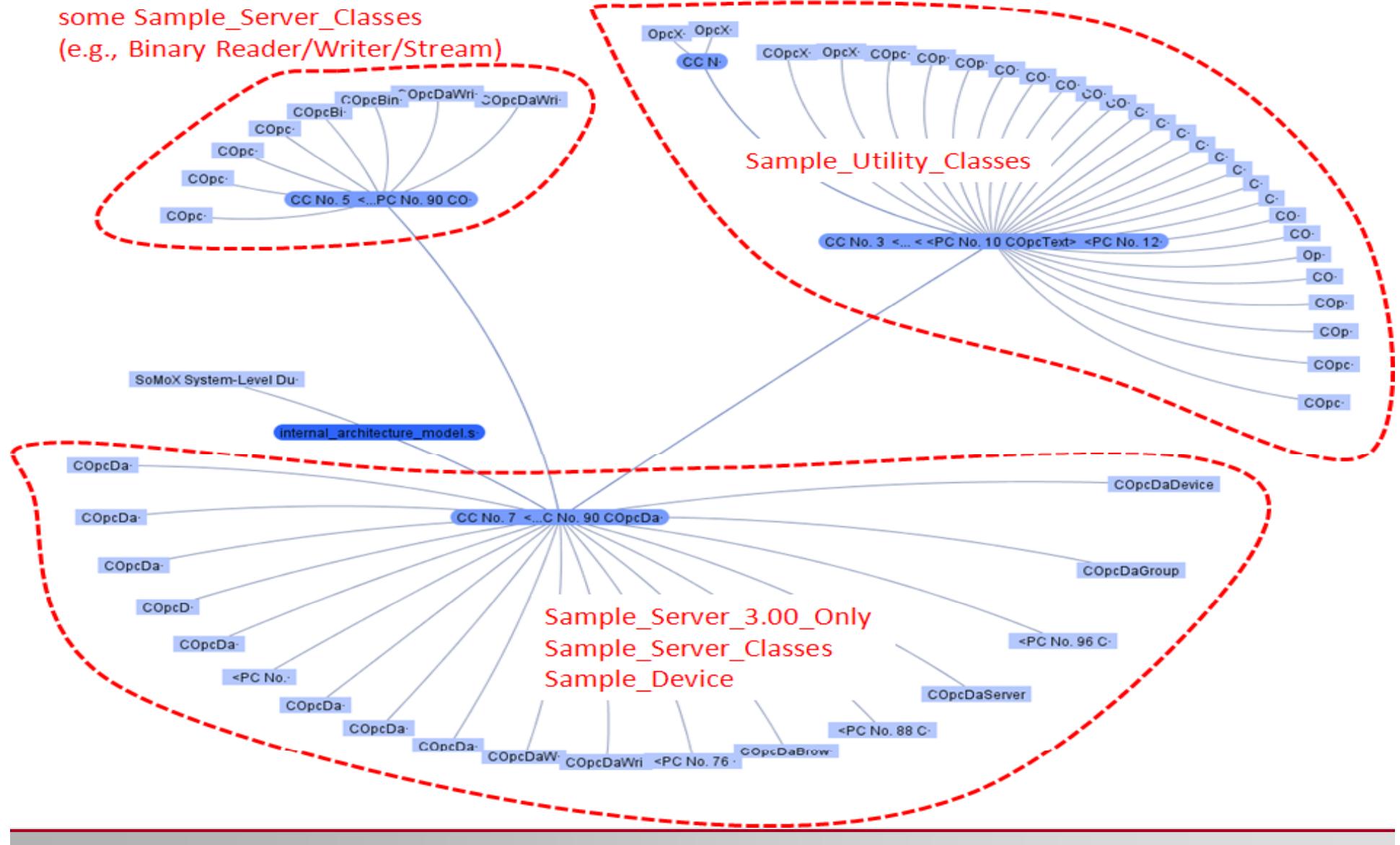


- Component-Based Software Architectures
- Component Behaviour
- Trace Model

Reverse engineering tools



Reverse engineering visualization



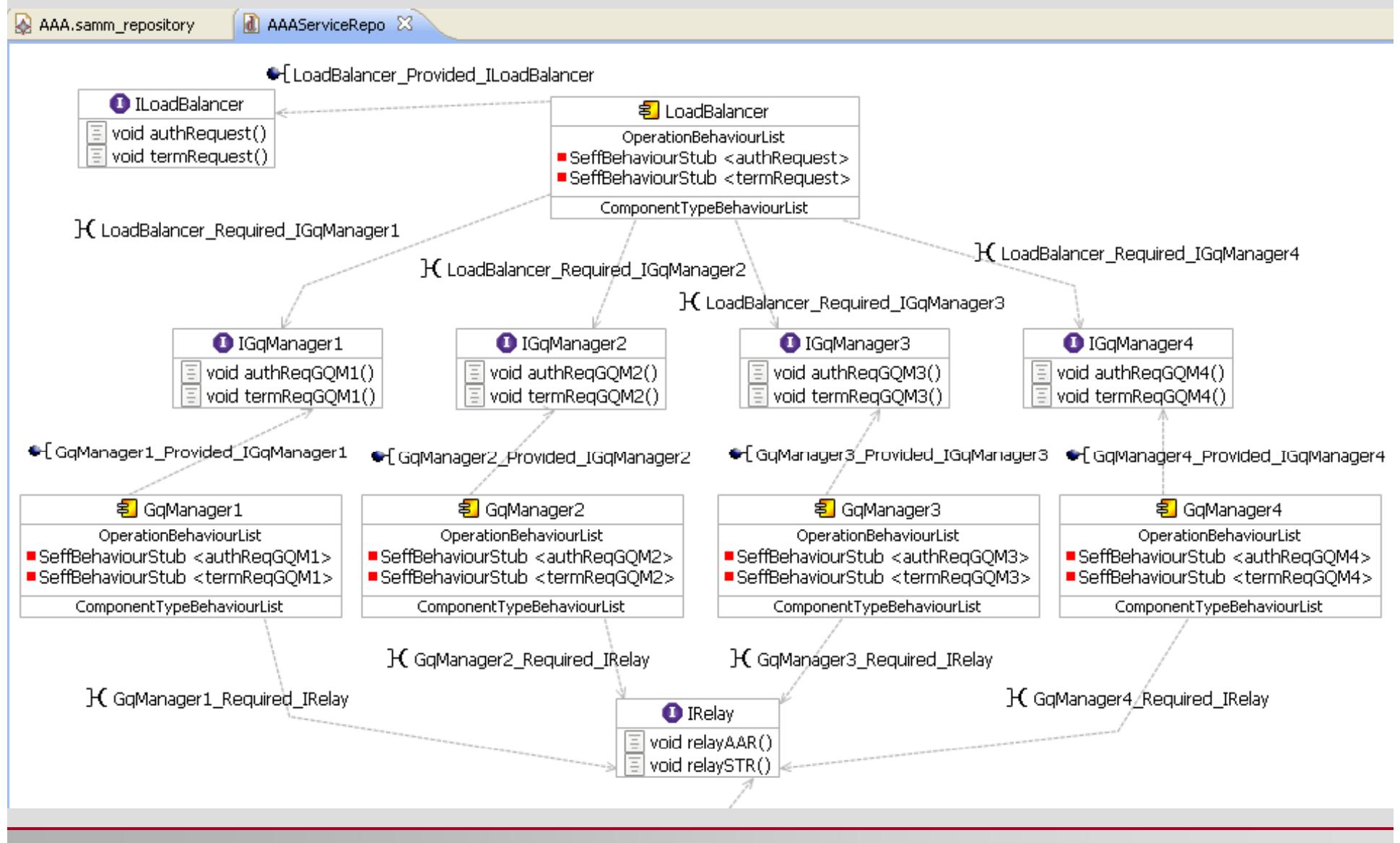
Outline

- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Modelling tools

- **UML-like modelling for all models**
 - GUI modelling (also textual, if necessary)
 - Automatic GUI layout for reverse engineered models
- **Modelling of „what-if” scenarios**
 - Alternative(s) to some base model
 - Alternative(s) applicable to all models

Modelling tools



Modelling tools - Editors

- **Tree-based**

- Inherently provided for all models

- **Graphical**

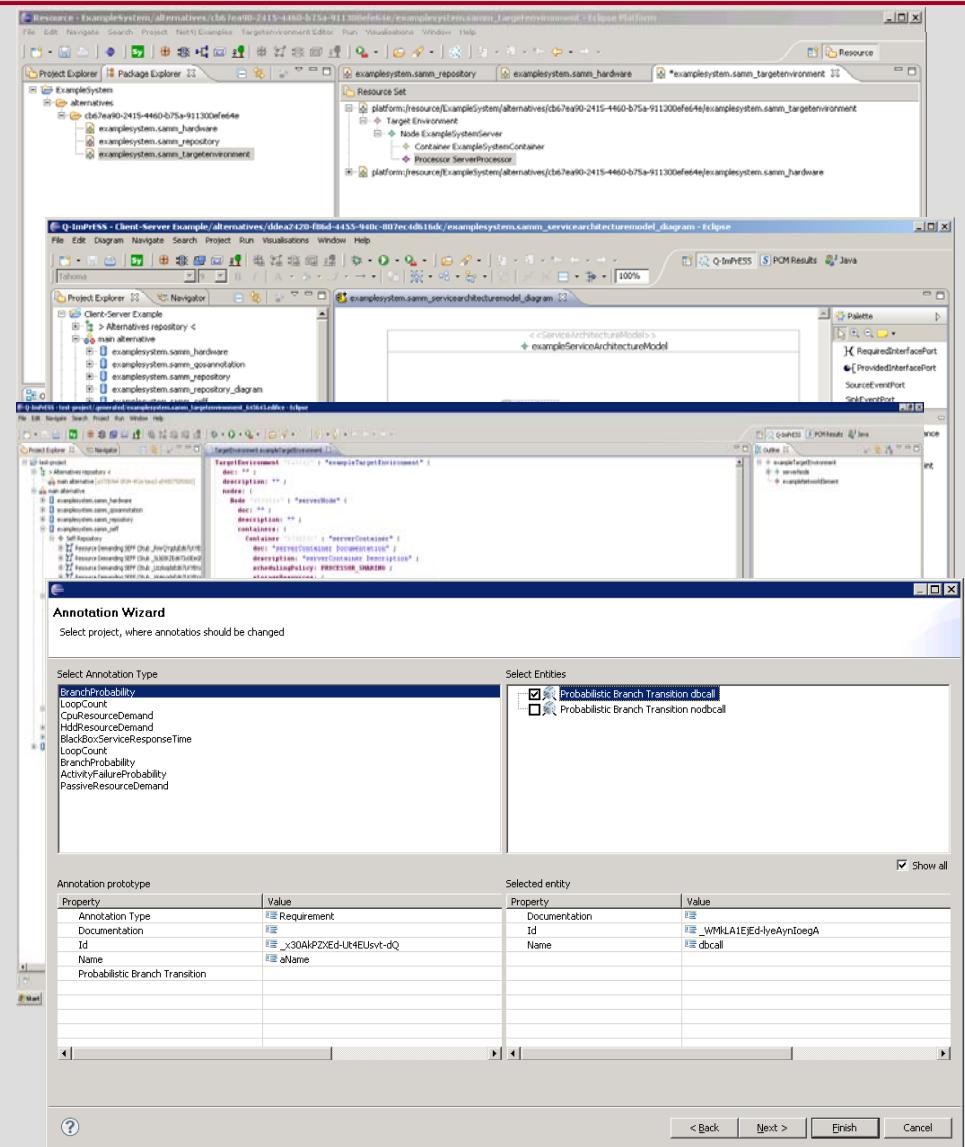
- To model static structure
- Provided only for particular models

- **Textual**

- To simplify editing of model properties

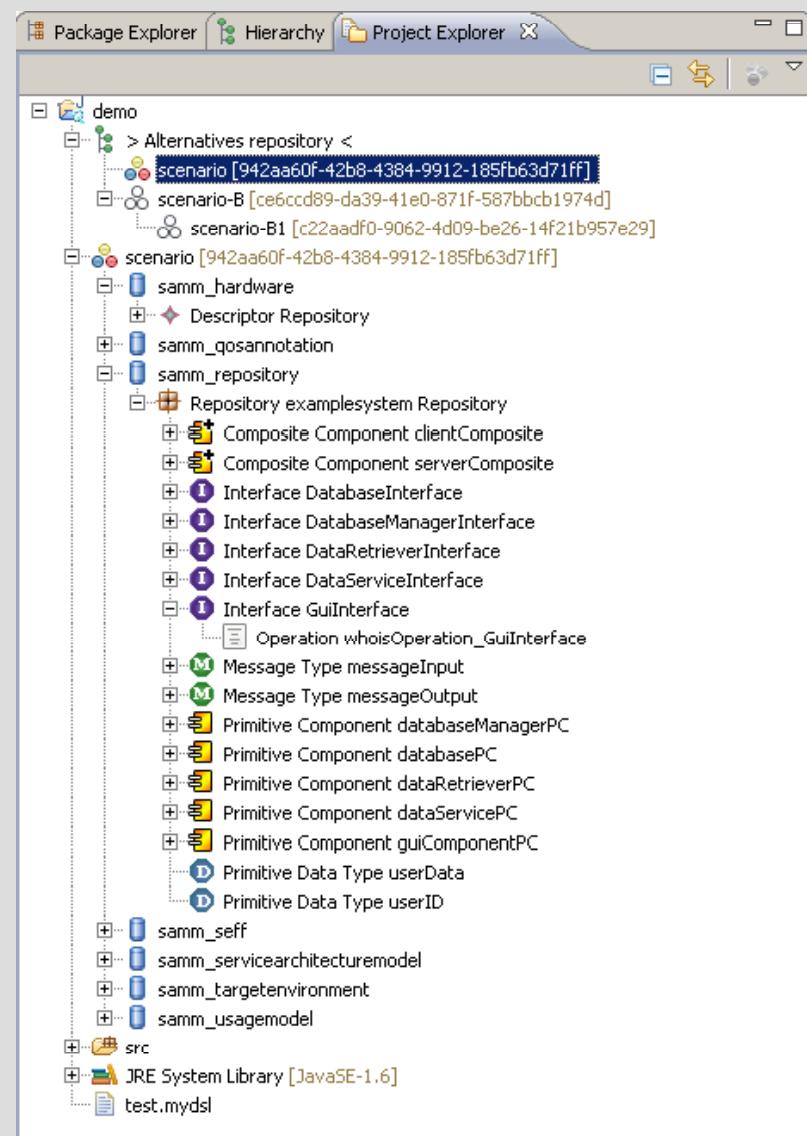
- **Ad-hoc**

- E.g., Quality annotation editor



Modelling tools – Alternatives concept

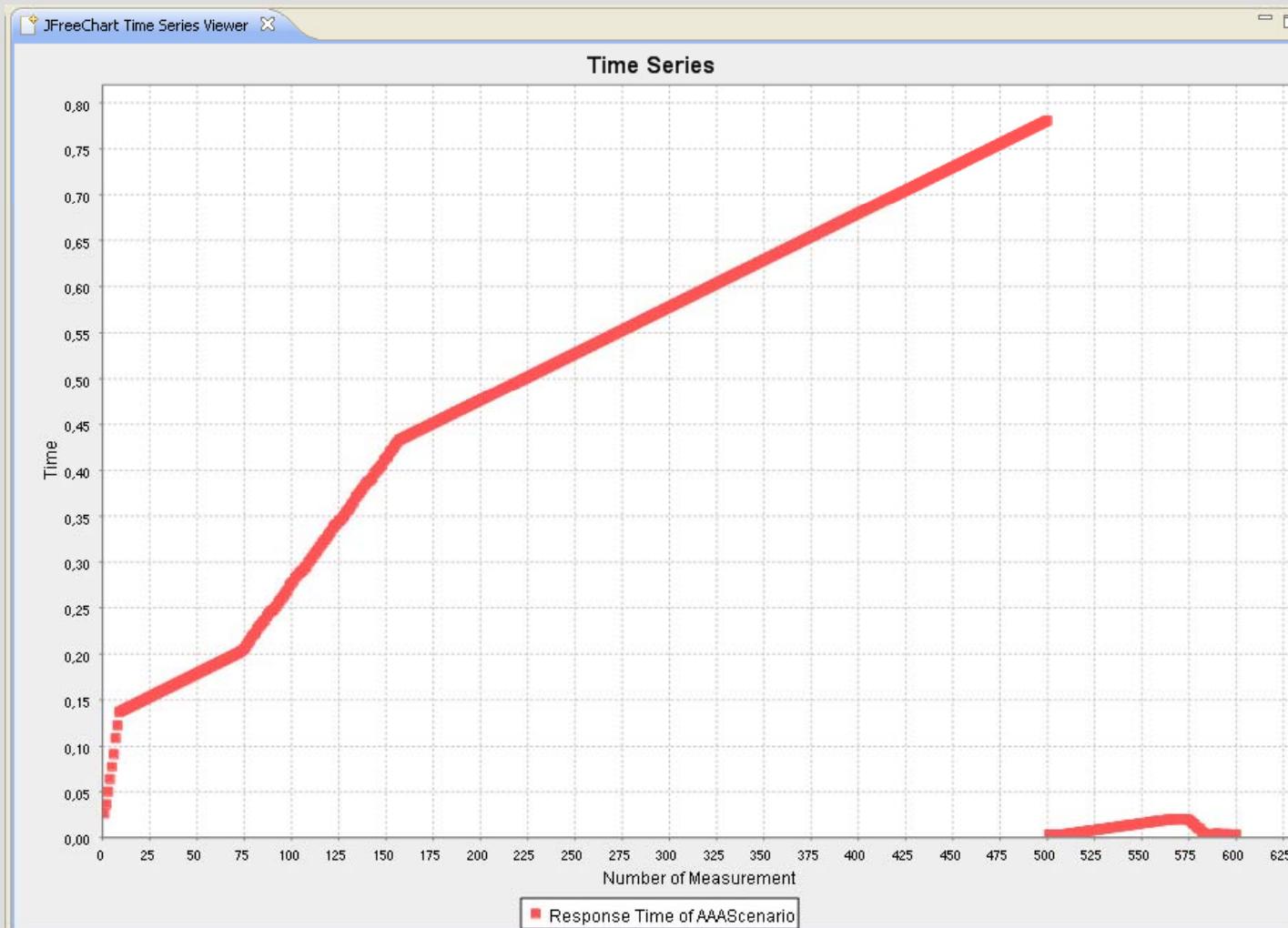
- **Reflects various modeling scenarios**
 - Base for model versioning
- **Hierarchical structure**
 - Simplify inheritance of models
- **Serves as input for analysis**
 - Performance
 - Trade-off
 - ...



Outline

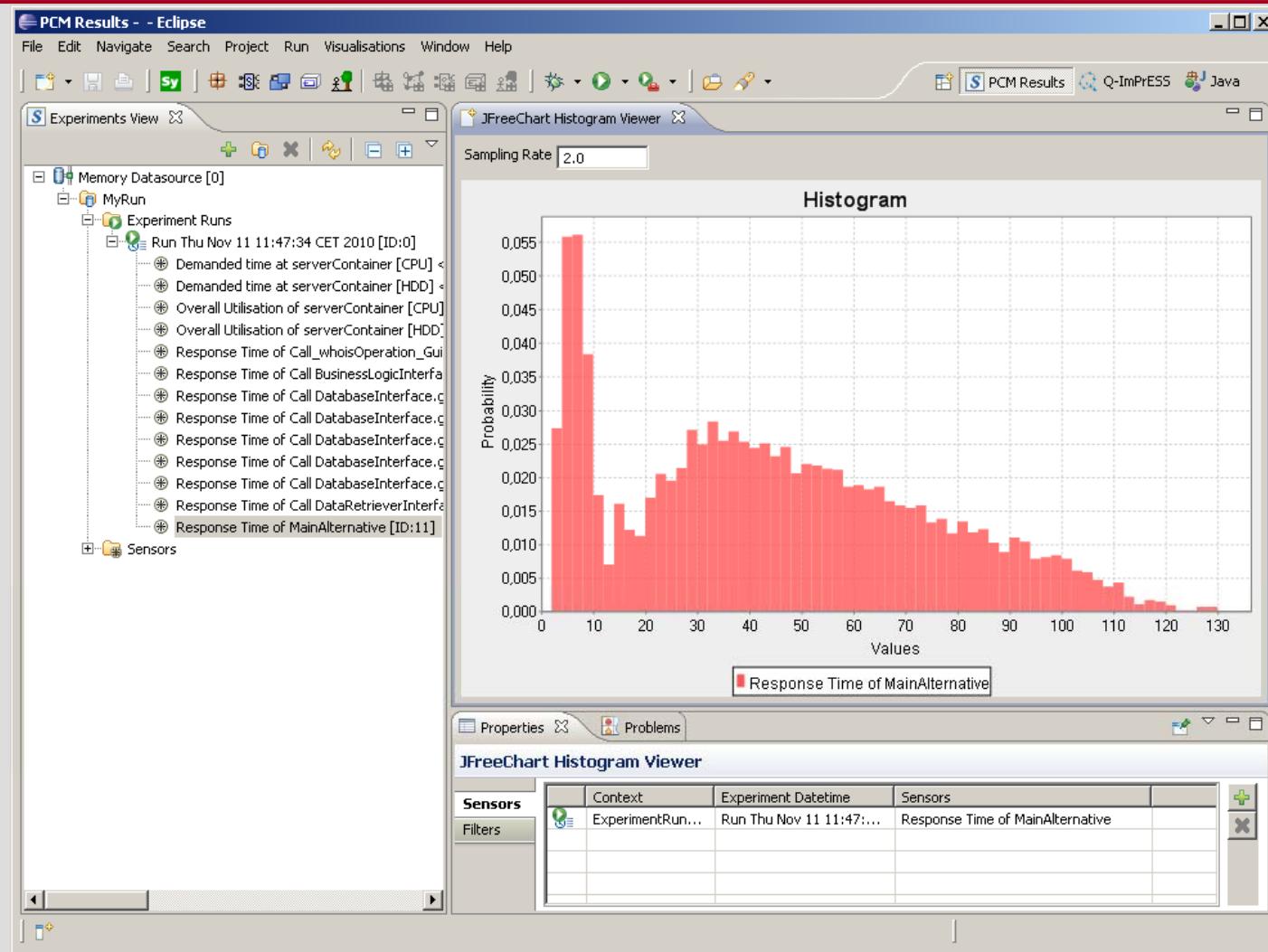
- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Prediction tools – Results example



Absolute response time prediction

Prediction tools – Results example

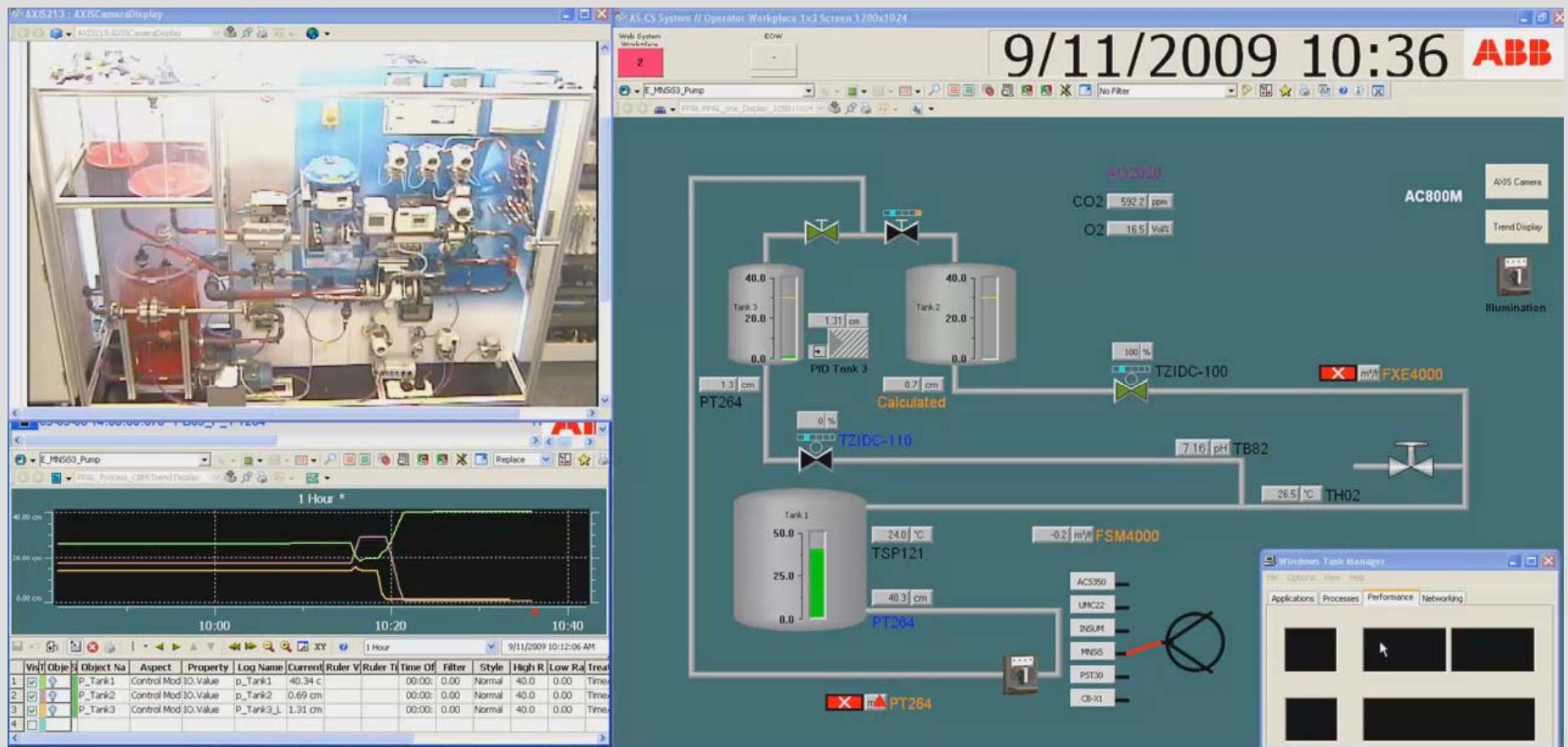


Distribution of response times

Outline

- **Q-ImPrESS**
 - Project information
 - Project organization
 - How to use project results
 - How to model a software system
 - Developed tools
 - Reverse engineering tools
 - Modelling tools
 - Prediction tools
 - Project validation

Q-ImPrESS validation



Q-ImPrESS validation

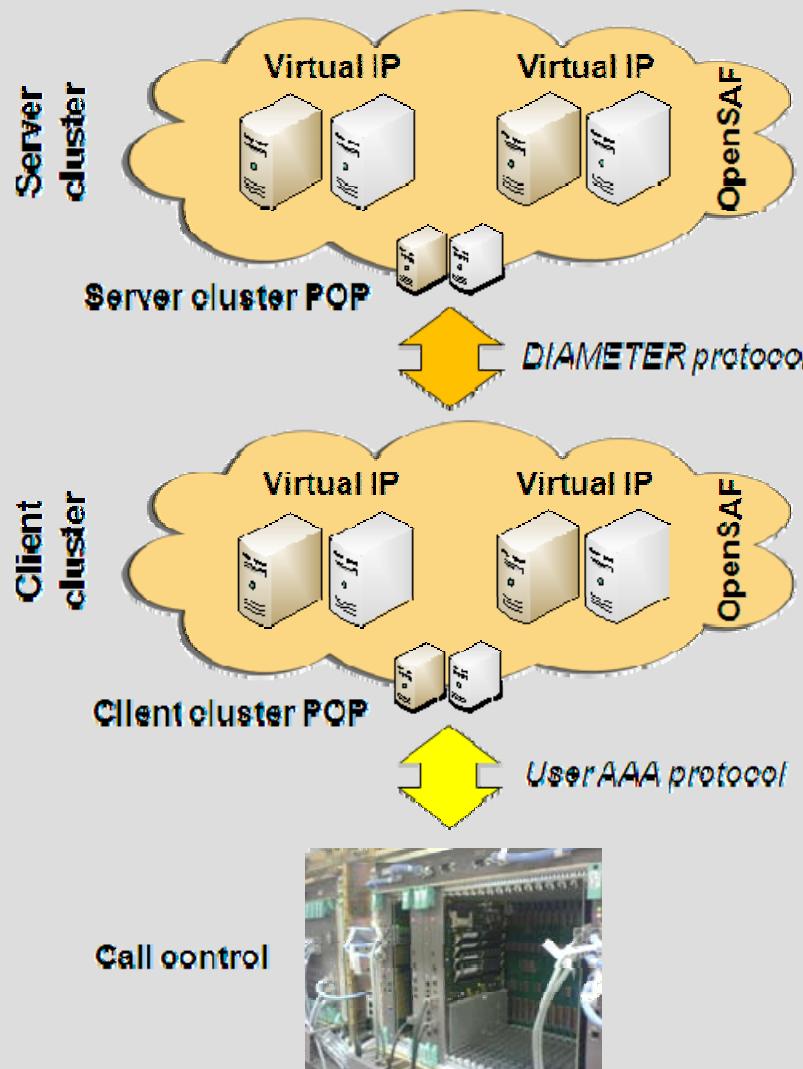
ABB



Q-ImPrESS validation



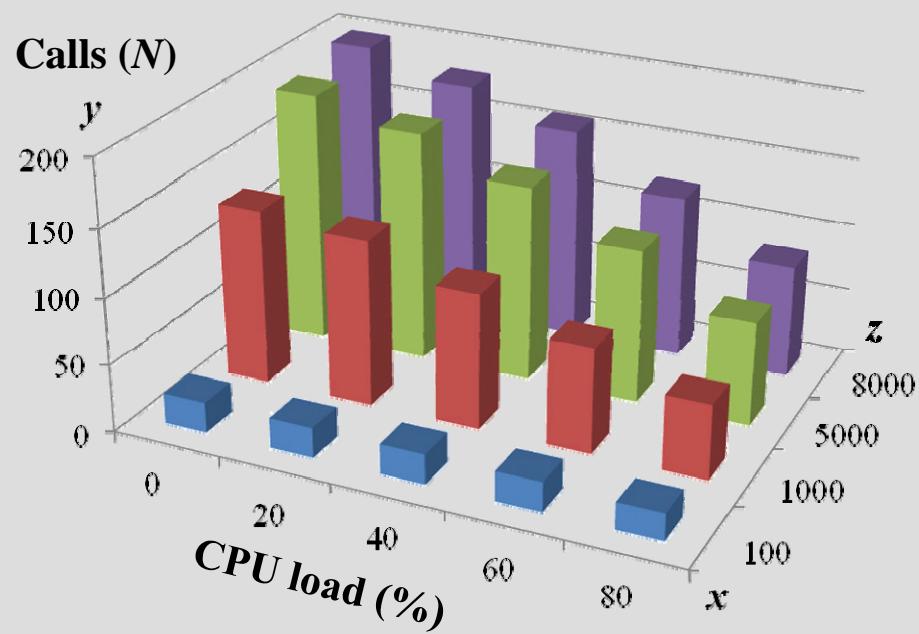
Q-ImPrESS validation



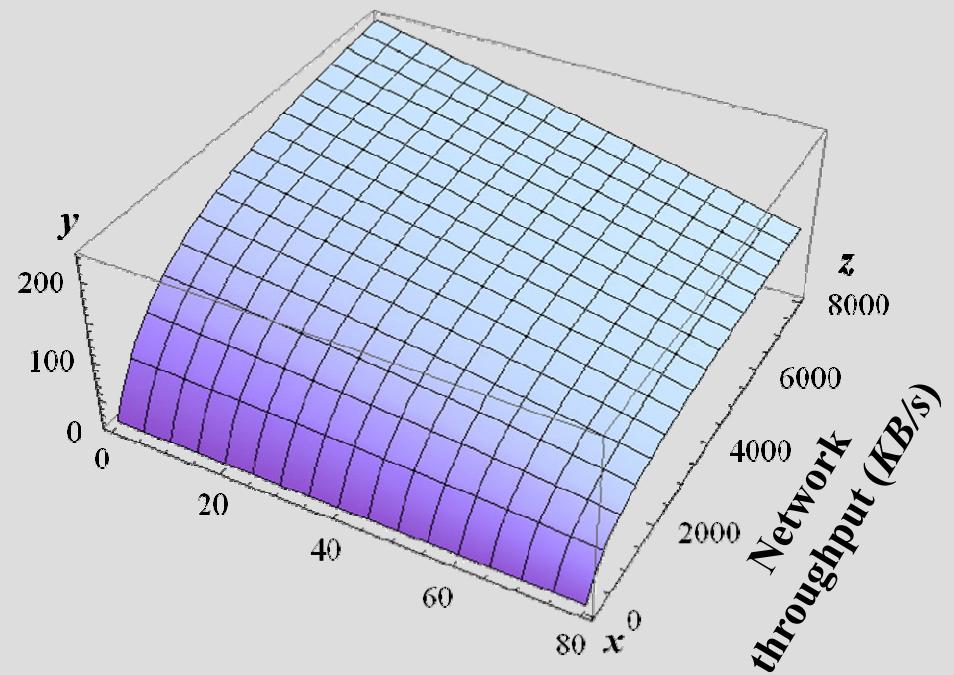
- Extension of the call control with AAA function based on widely-accepted DIAMETER standard
- Open-source software used
 - Linux (CentOS)
 - OpenSAF → **SERVICE AVAILABILITY FORUM™**
 - OpenDIAMETER
 - Pen (TCP level load balancer)
- Availability and reliability
 - Assured through OpenSAF
 - High availability middleware

Q-ImPrESS validation

Experimental results



Model prediction



Final status

- **Three successful reviews from EU**
 - Highest possible grade
 - One of the most successful FP7 projects
- **50 deliverables delivered**
 - Several millions lines of code
 - More than 2000 pages of text
- **Plans for future projects and research**