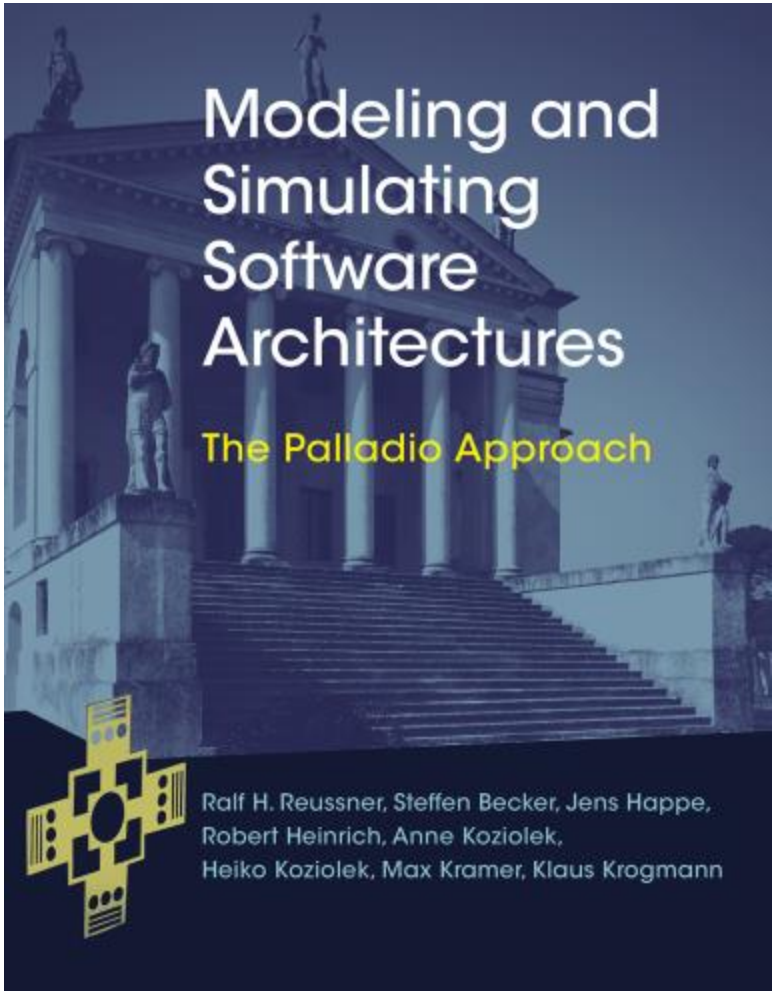
ICSA 2017 Tutorial
Runtime Modeling and Visualization
--
Introduction to Palladio

Schedule of Events

09:00 – 09:10	Welcome and General Introduction
09:10 – 09:40	Study Foundations
09:40 – 10:00	Model-based Software Application Monitoring
10:00 – 10:30	Runtime Architecture Modeling and Visualization
10:30 – 11:00	Coffee Break
11:00 – 12:15	**Introduction to the ExplorViz, Palladio, and iObserve Approaches with following Tool / Visualization Demos**
12:15 – 12:30	Study Setup
12:30 – 14:00	Lunch
14:00 – 15:30	Comprehensibility Study
15:30 – 16:00	Coffee Break
16:00 – 16:30	Live Database Trace Visualization in Large Software Landscapes
16:30 – 17:00	Feedback and Open Discussion

Palladio Book



By Ralf H. Reussner, Steffen Becker, Jens Happe, Robert Heinrich, Anne Koziolek, Heiko Koziolek, Max Kramer and Klaus Krogmann

400 pp., 98 illus.
ISBN: 9780262034760

MIT Press, Cambridge, MA, October 2016.

Overview

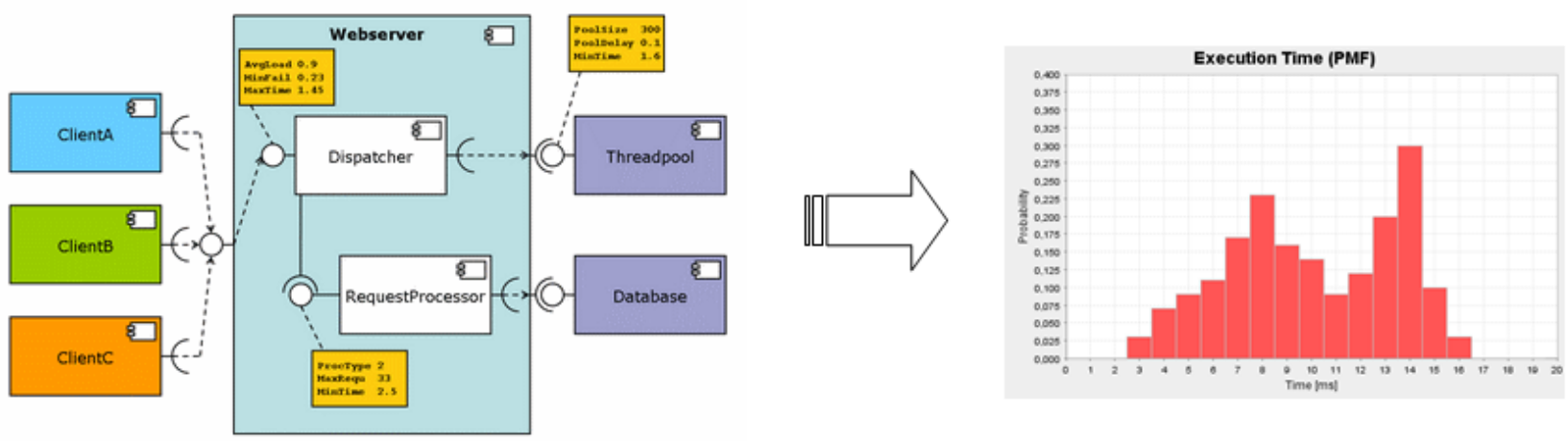
- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- Roles in Component -based Software Development
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Overview

- **Introduction**
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- Roles in Component -based Software Development
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Introduction

- Prediction of quality properties on a model base
  - for systematic design of software systems
  - performance, reliability, costs



- Derive performance metrics from the models using
  - analytical techniques and
  - simulation

Overview

- Introduction
- **Palladio**
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- Roles in Component -based Software Development
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Andrea Palladio

- Andrea Palladio (30 November 1508 – 19 August 1580) was an Italian architect active in the Republic of Venice.
- Palladio, influenced by Roman and Greek architecture, primarily by Vitruvius, is widely considered the most influential individual in the history of Western architecture.
- All of his buildings are located in what was the Venetian Republic, but his teachings, summarized in the architectural treatise, The Four Books of Architecture, gained him wide recognition.

[Wikipedia]

Palladio

- *Palladio is an approach for the definition of software architectures with a special focus on performance properties.*

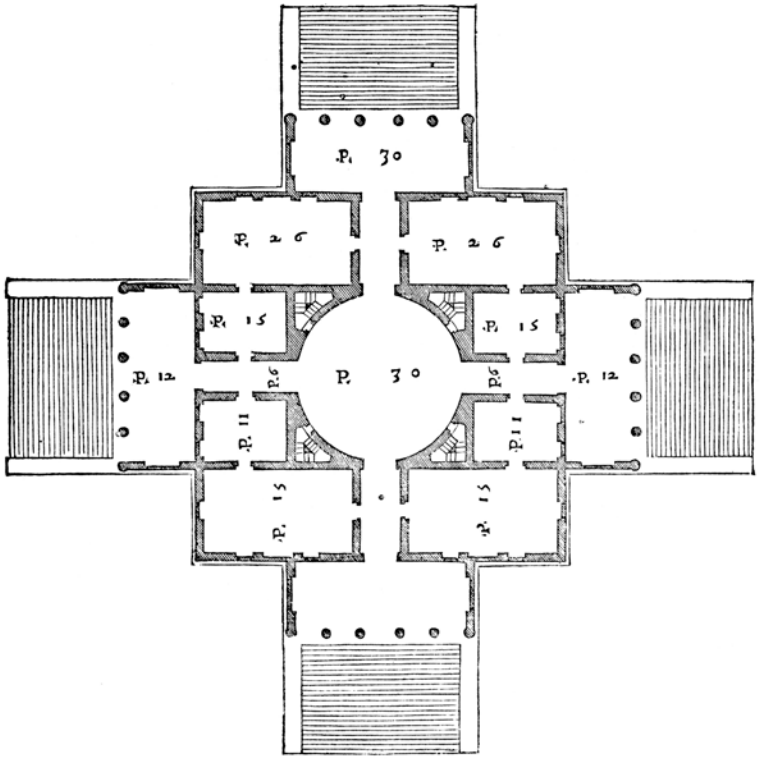
[Reussner 2016a]

- The Palladio Component Model
  - as *one example* for a conceptually clear component model
  - gives an overview on component concepts
  - defines a process view

Overview

- Introduction
- Palladio
- **Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach**
- Roles in Component -based Software Development
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Palladio Component Model (1)



- The Palladio Component Model (PCM) is designed to enable early performance predictions for software architectures
- Aligned with a component-based software development process
- Targets at
  - Performance prediction for component-based software architectures
  - Business information systems

Palladio Component Model (2)

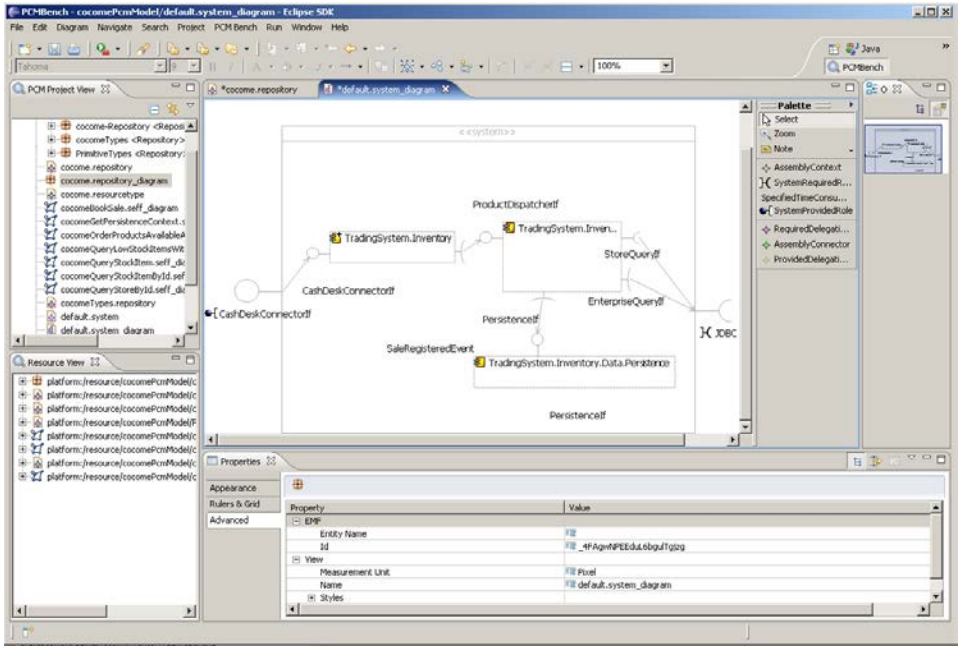
- Enables developers to create PCM model instances with graphical editors
- Derives performance metrics from the models using
  - Analytical techniques and
  - Simulation
- Development started in 2003
- Model is named after famous Italian Renaissance architect Andrea Palladio (1508-1580)
- Extensive metamodel in EMF/Ecore

Application of PCM

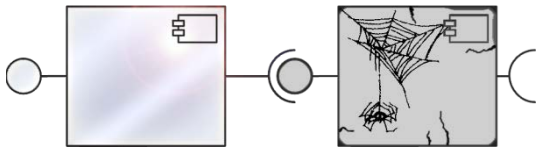
- Systematic design of software systems
- Use of component-based technologies
- Focus: Quality properties
  - In particular: Performance
  - Utilises model-driven performance prediction approaches

Tool: PCMBench

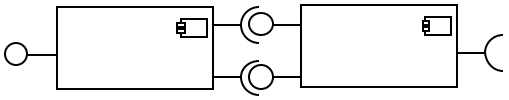
- Supports the whole component-based design process
- Analysis approaches provide hints on performance bottlenecks / issues



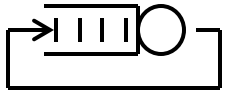
Why Creating Models?



Extensions of legacy software systems



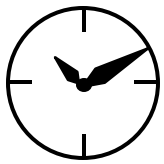
Performance predictions at design time



$$\sum_{i=0}^{N} p_l(i) \left(\circledast_{j=1}^{i} f_P \right)(t)$$

Analytical solvable

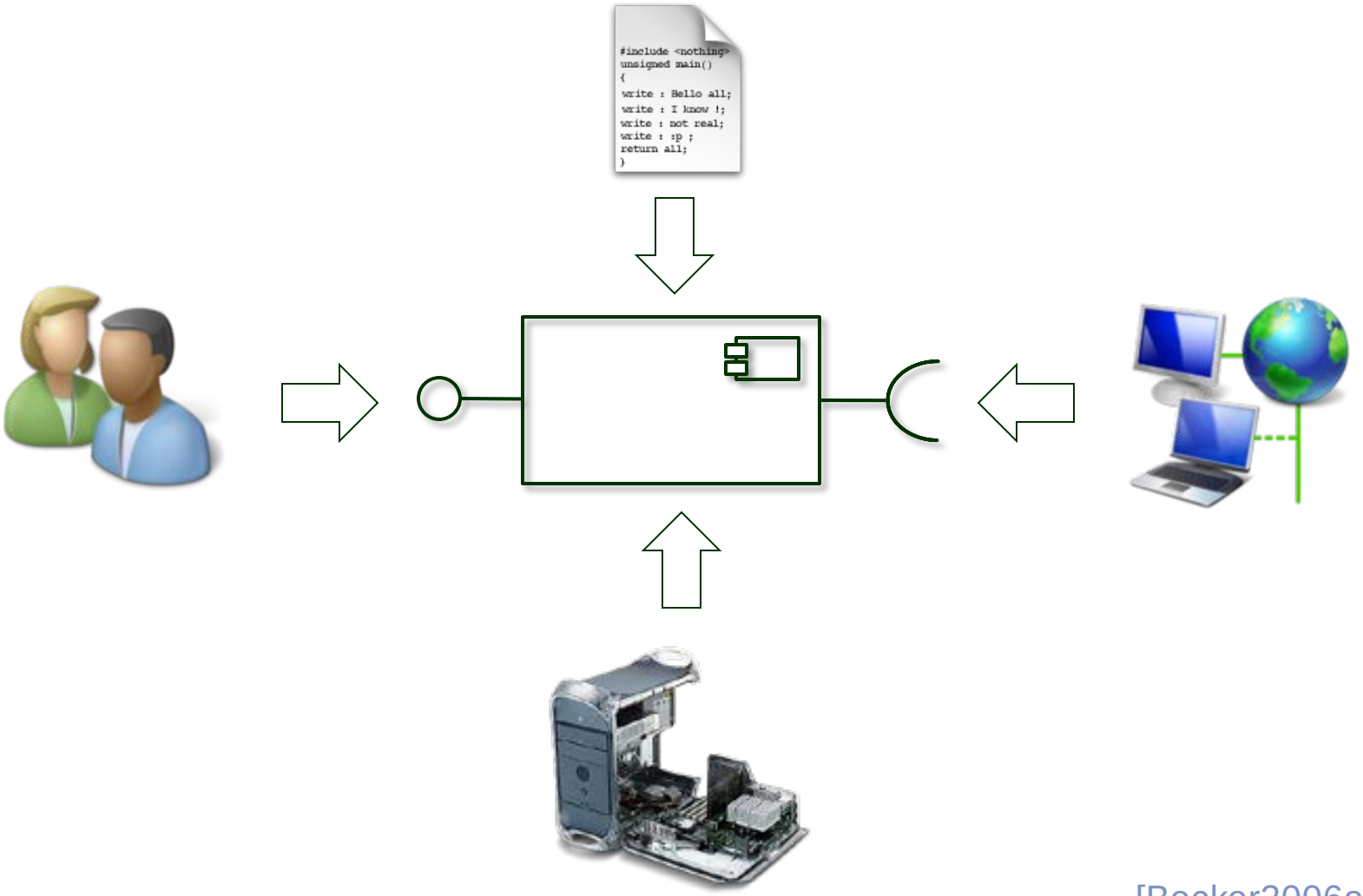
Performance jumps
Critical contention levels



Simulation

Faster than “real” execution

Component Performance (1)



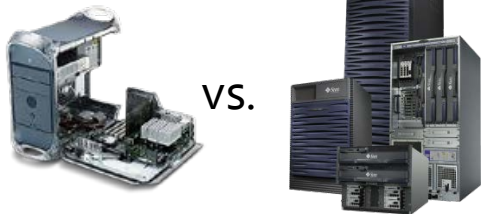
[Becker2006a]

Component Performance (2)

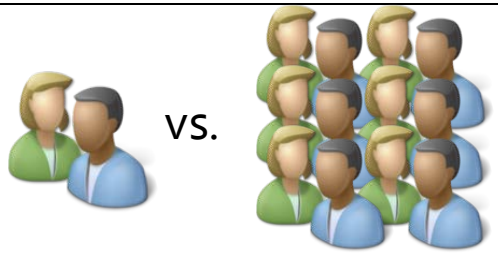
- *All* influence factors are made explicit in the PCM
- Required for conceptually clear components
- Supported context changes:
  - Allocation context → execution system (hardware / middleware / virtual machines)
  - Usage context → usage profile
  - Assembly context→ “wiring” (other components fulfil a required services)

  → Explicit parameters in the PCM

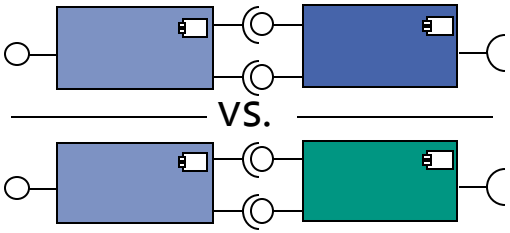
Supported Context Changes



Changing hardware
Sizing / scalability / relocation



Changes in usage profile
Users interact differently with the system

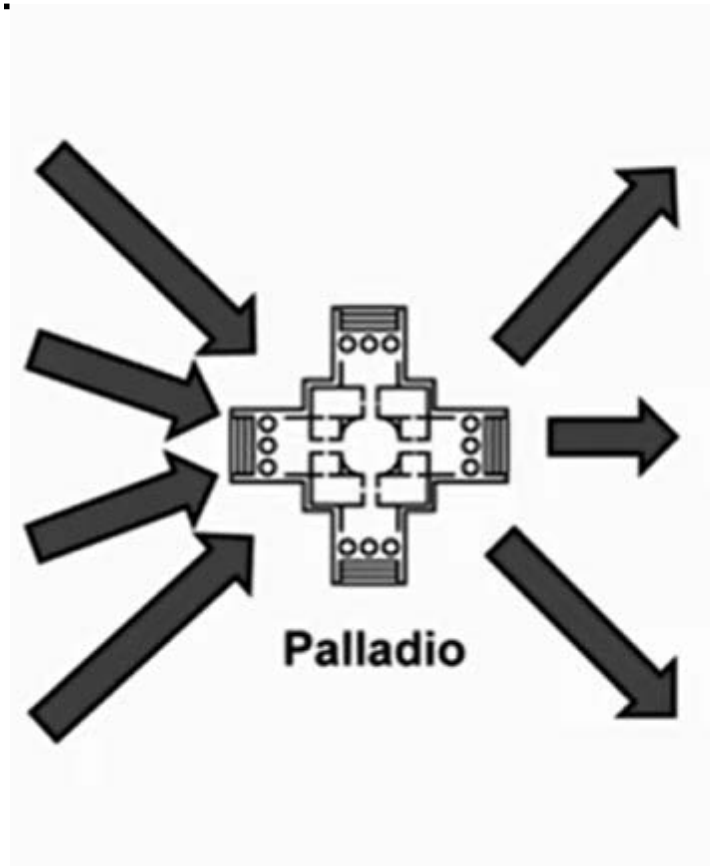


Changes in assembly context

Component Description

What are the intuitive inputs and outputs for a performance prediction model?

?



?

also see: http://www.youtube.com/watch?v=H0Gj-kdG[illegible]s

Overview

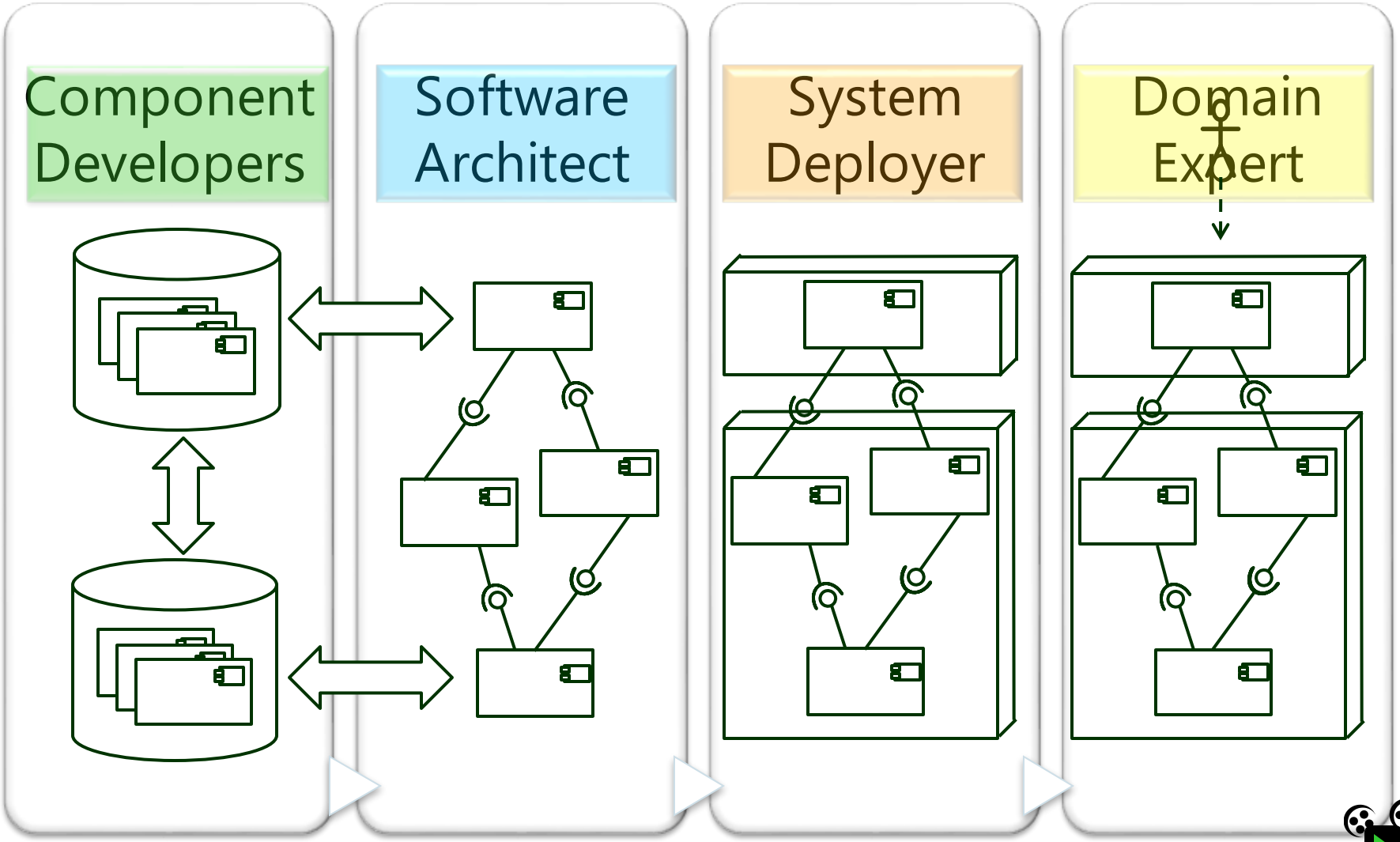
- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- **Roles in Component-based Software Development**
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Idea

- Component Based Software Engineering (CBSE) principles require
  - Third party use
  - Readily composable
  - → Separate developer roles
- Component performance depends on
  - External services
  - Resource environment
  - Usage
  - → SEFF and parameterisation

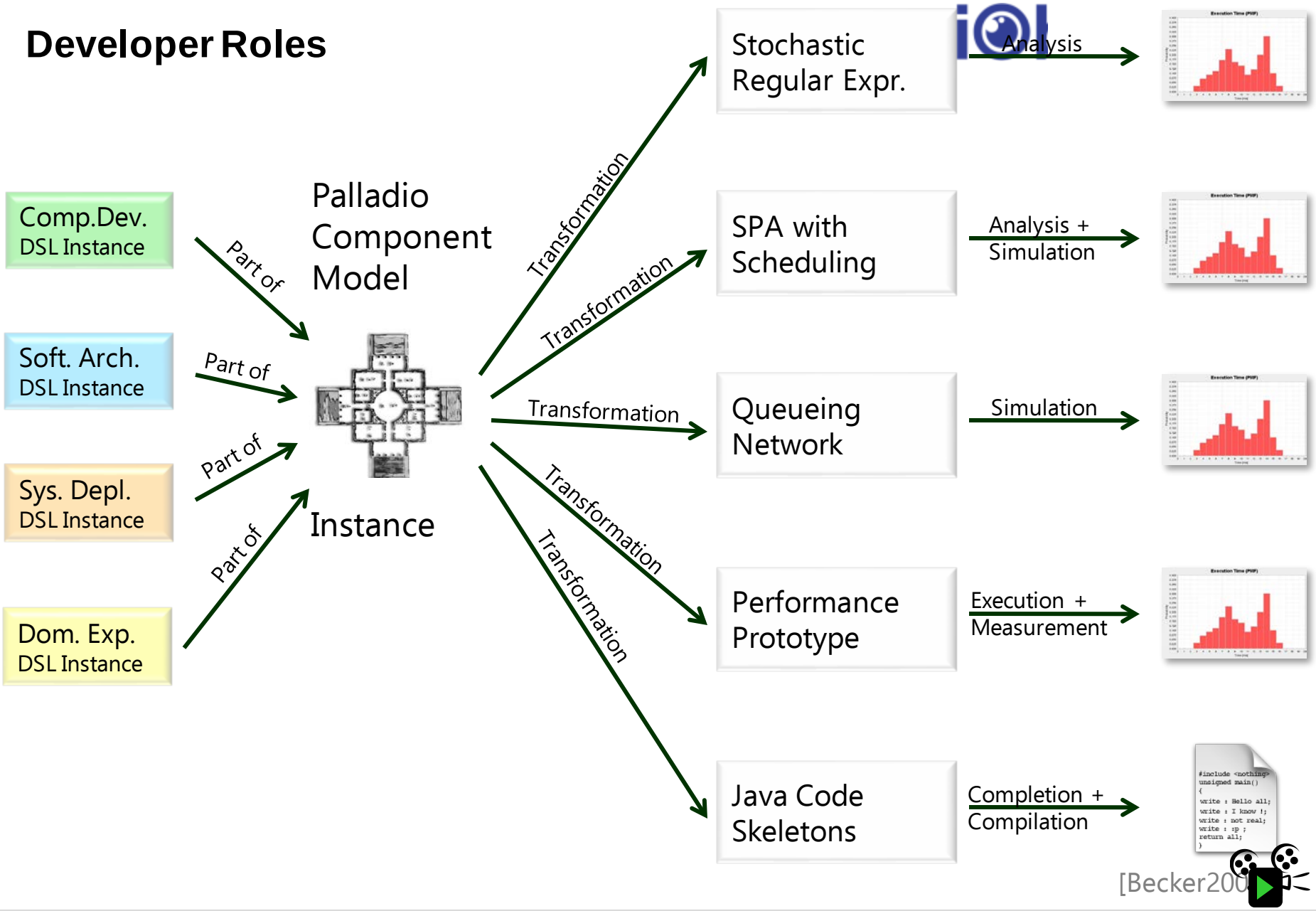
Developer Roles

Developer Roles



[Becker2007a]

Developer Roles



[Becker200

Overview

- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- **Roles in Component -based Software Development**
  - **Component Developer**
  - Software Architect
  - System Deployer
  - Domain Expert
- Conclusion / Summary

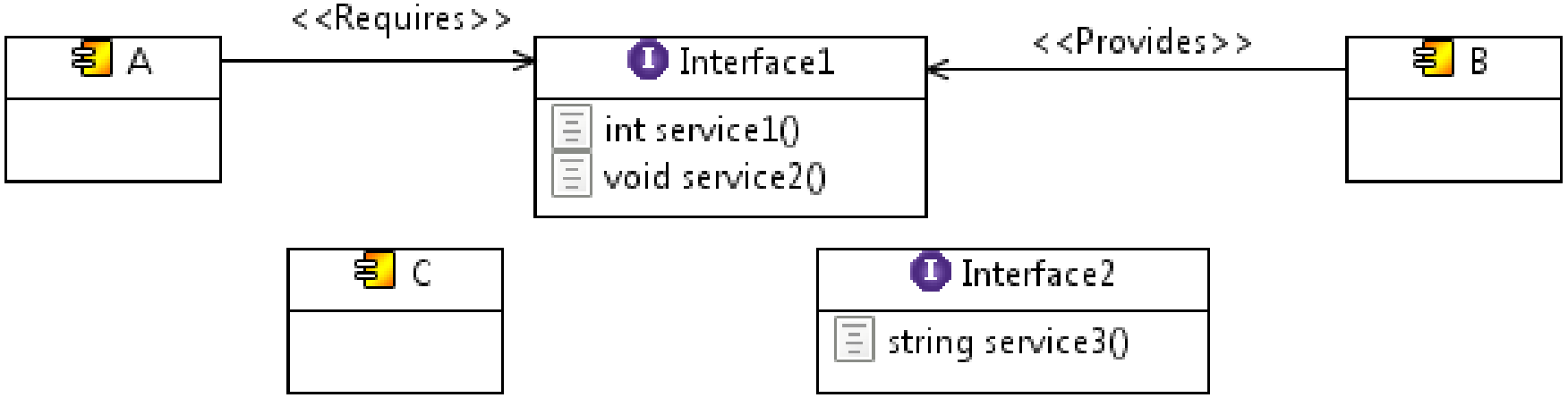
Component Developer: Tasks

- Specifies components & interfaces
- Specifies data types
- Builds composite components
- Creates service effect specifications
- Stores modelling & implementation artefacts in repositories
- Implements components
- Tests components
- Maintains components

Component Developer

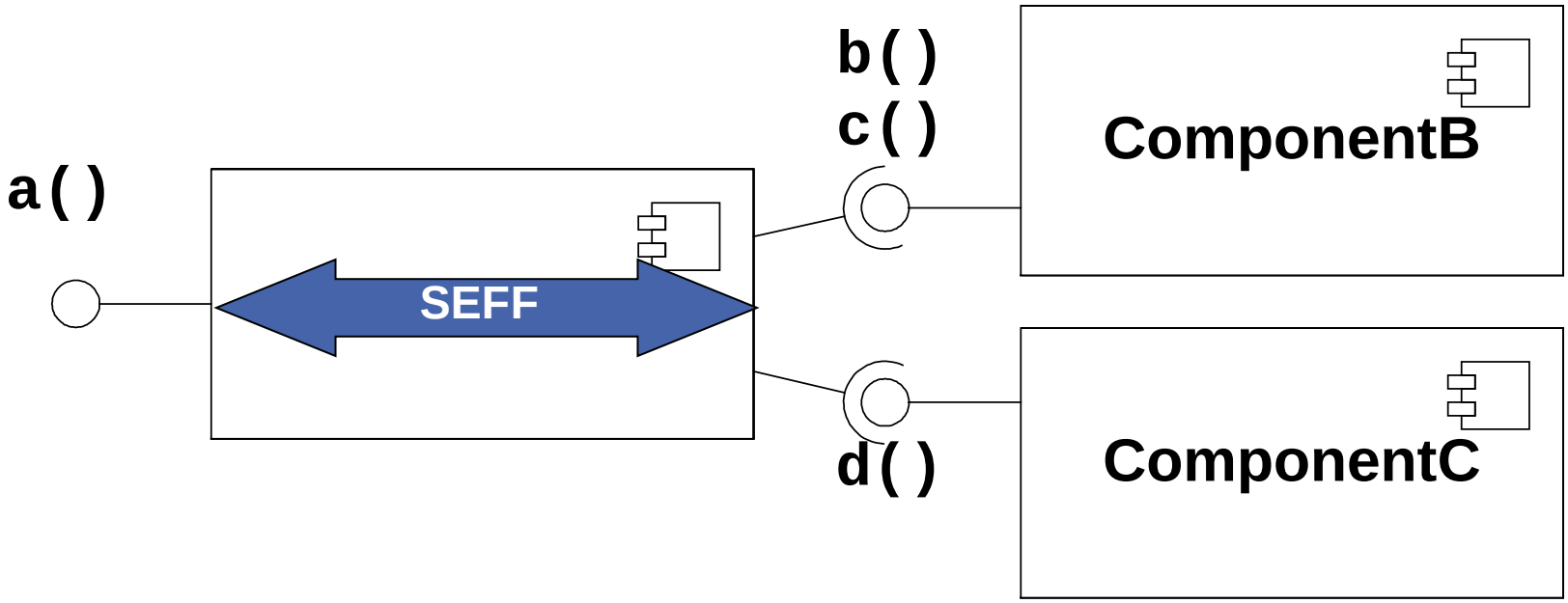
Example: Component Description in Palladio

- Component interfaces need to be described
- Created components are stored in a repository



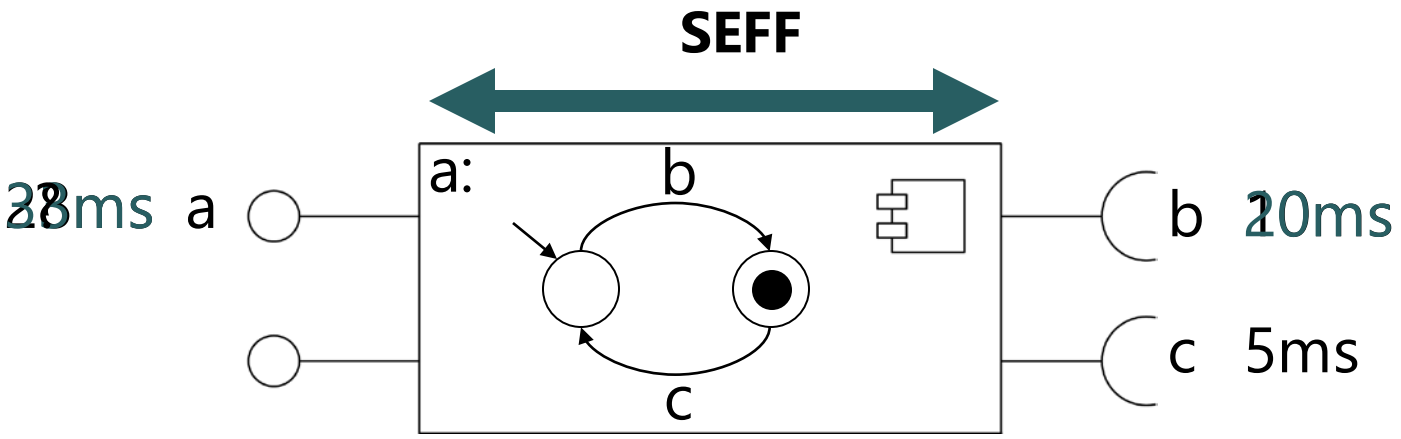
Component Developer

Example: Service Effect Specification: Idea



Component Developer

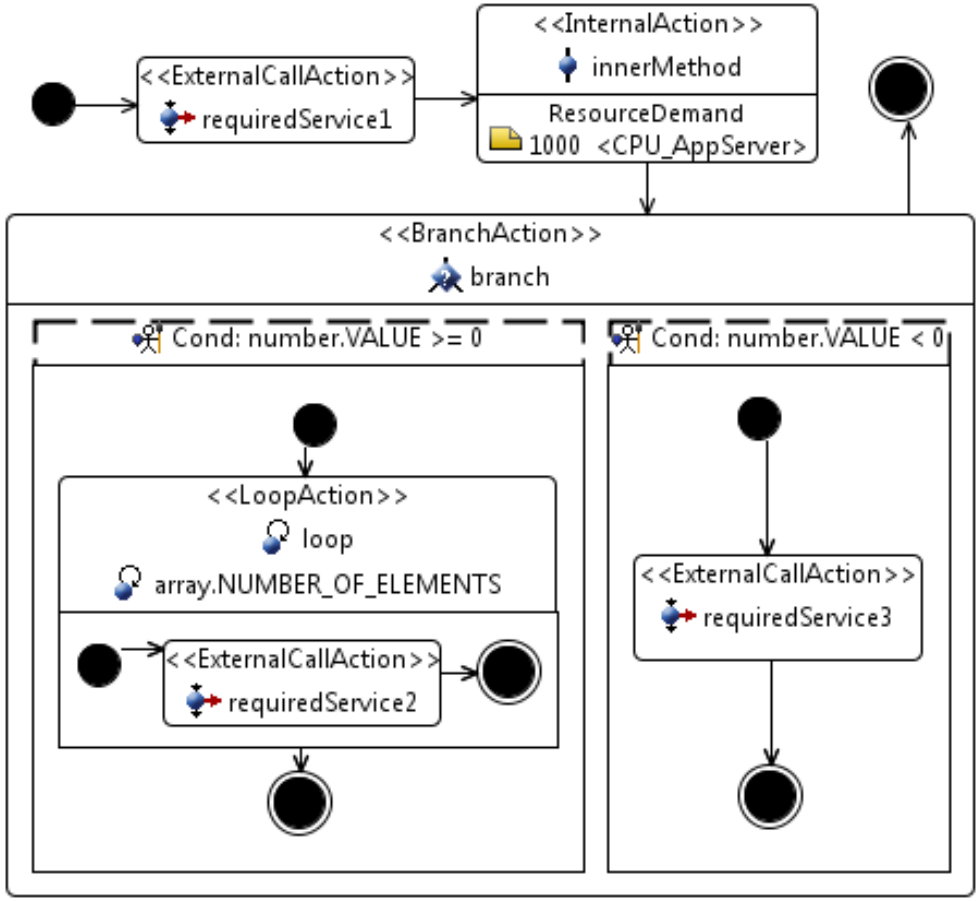
Example: Service Effect Specification: Idea



Component Developer

Example: Service Effect Specification in Palladio

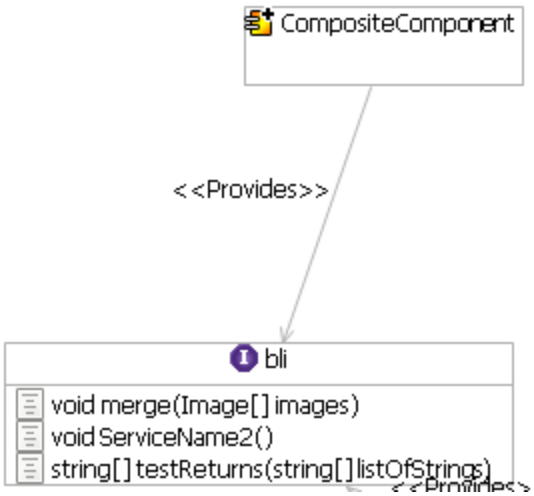
- Component behaviour needs to be described in so-called Service Effect Specification (SEFF)

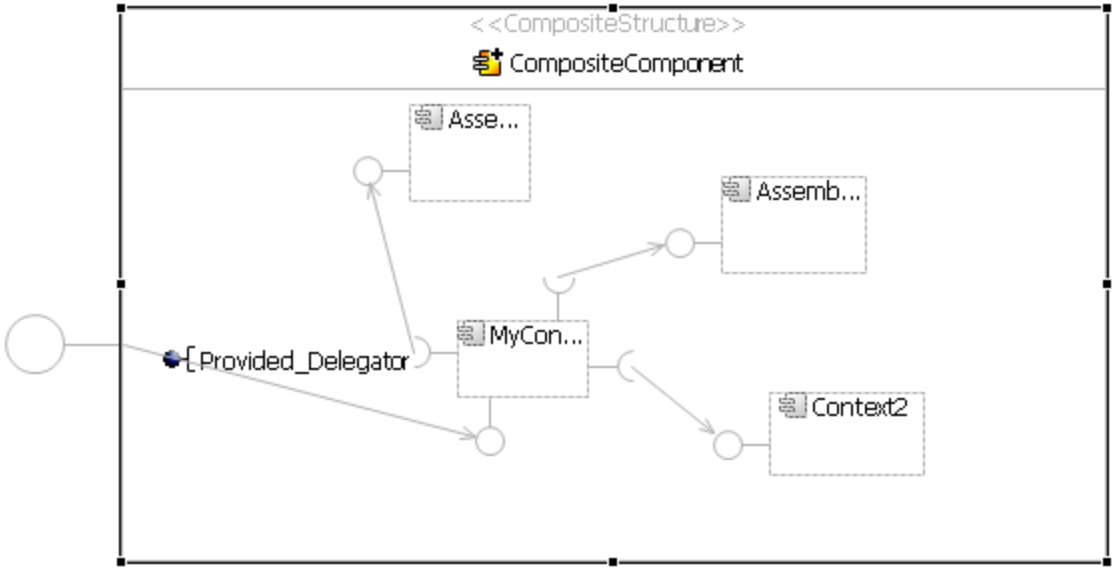


Component Developer

Example: Composite Component in Palladio

- Composed from Basic Components and/or other Composite Components





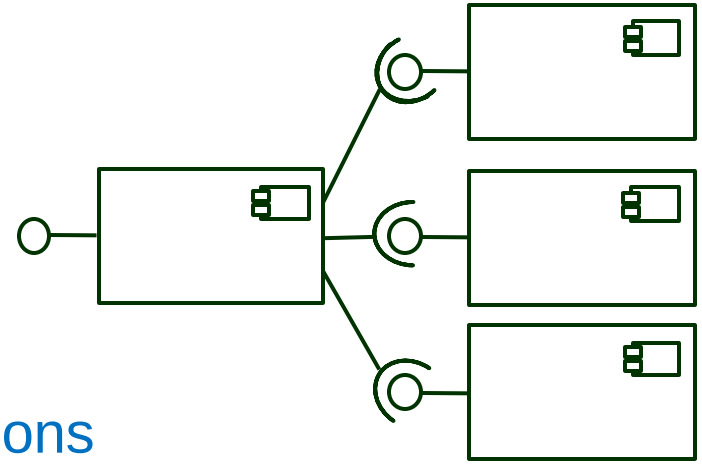
Component Developer

Overview

- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- **Roles in Component-based Software Development**
  - Component Developer
  - **Software Architect**
  - System Deployer
  - Domain Expert
- Conclusion / Summary

Software Architect: Tasks (1/2)

- Specifies an architecture (boxes and lines) from existing components and interfaces
- Specifies new components and interfaces
- Uses architectural styles and architectural patterns
- Analyses architectural specification and makes design decisions



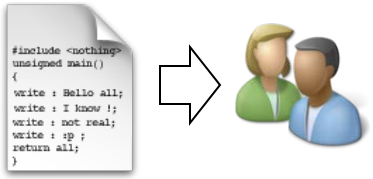
Software Architect

Software Architect: Tasks (2/2)

iObserve

- Conducts performance prediction based on architectural specification

- Delegates implementation tasks to component developers



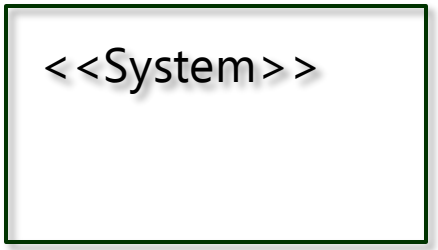
- Guides the whole development process

Software Architect

KIT Karlsruhe Institute of Technology

System Model

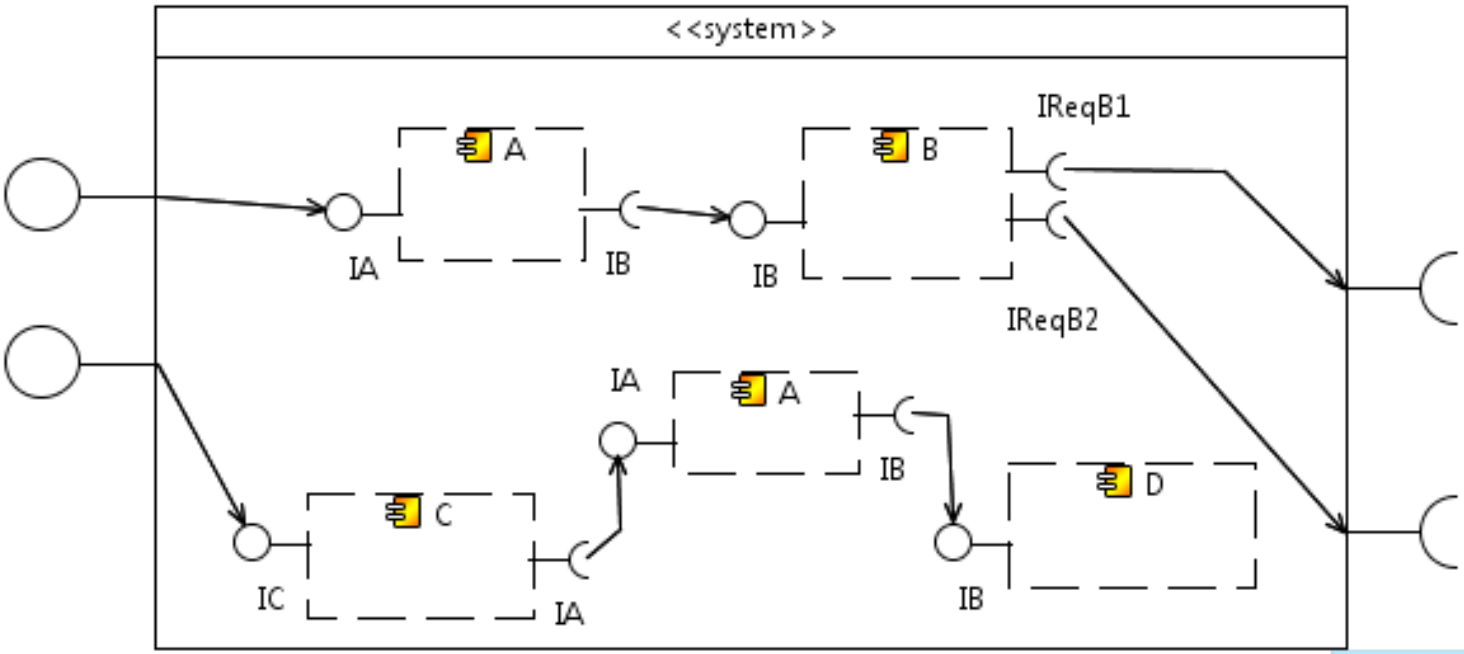
- Models the component-based architecture to be analysed
- May include components from different repositories
- Provides an interface for users
- Excludes uninteresting services and connects to them via system required interfaces
- Is a prerequisite for the system deployer to allocate the components



Software Architect

Example: System Composition in Palladio

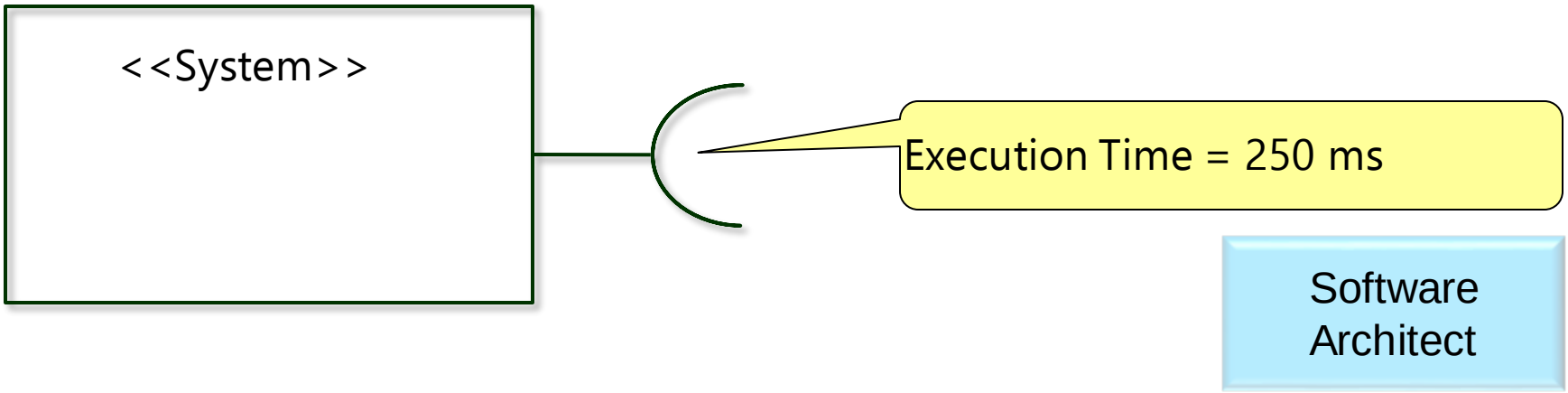
- System is composed of components from repository



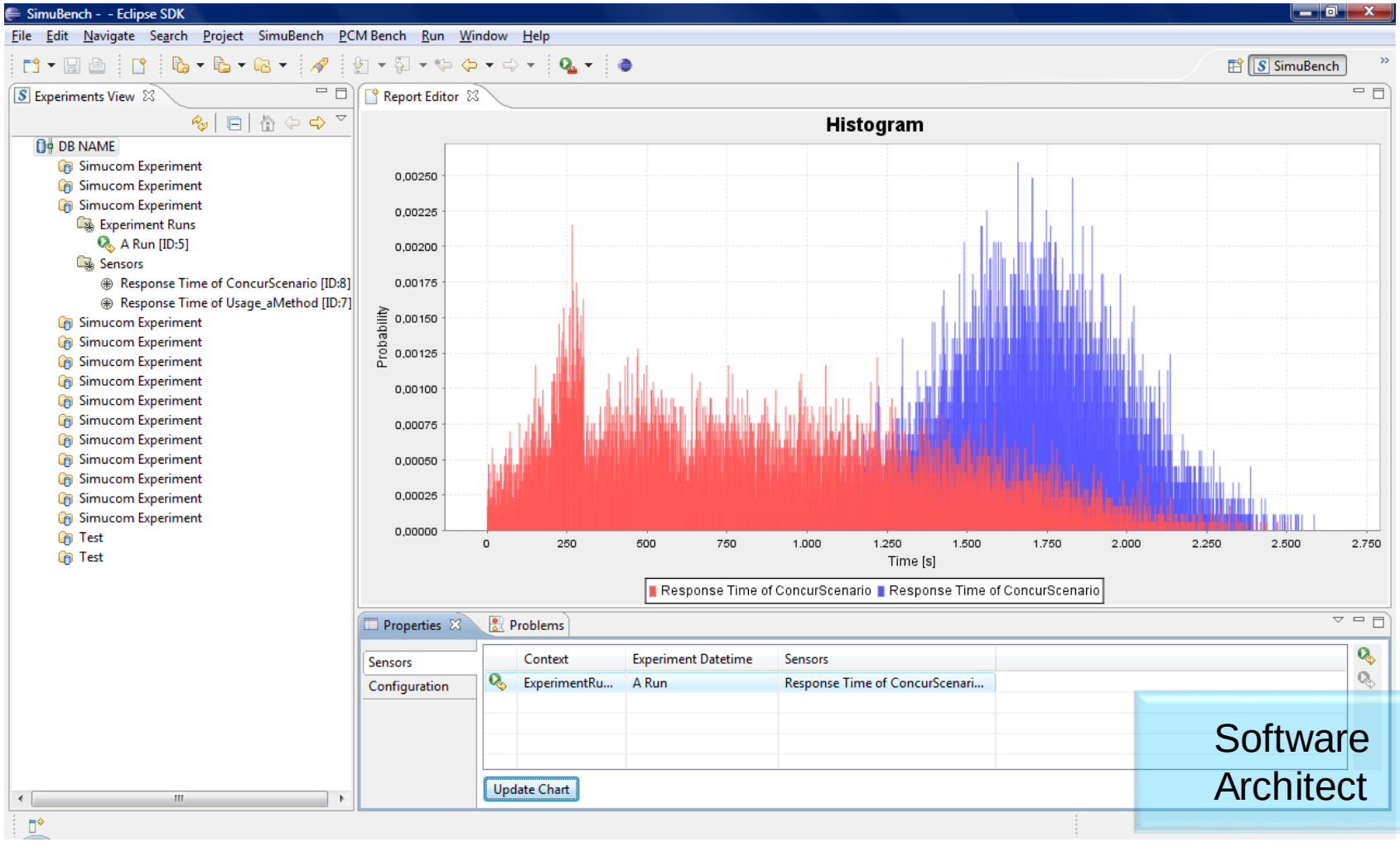
Software Architect

QoS Annotation

- System Required Interfaces: connection to functionality not modelled in the system
- Example: web service, unknown component
- Execution time specification necessary



Example: Performance Evaluation in Palladio



Overview

- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- **Roles in Component-based Software Development**
  - Component Developer
  - Software Architect
  - **System Deployer**
  - Domain Expert
- Conclusion / Summary

System Deployer: Tasks

- Models the resource environment (e.g., middleware, OS, hardware)
- Models the allocation of components to resources
- Sets up the resource environment (e.g., installing application servers, configuring hardware)
- Deploys components on resources (e.g., writing deployment descriptors)
- Maintains the running system

System Deployer

Resource Types

- Abstract specification of resources (e.g. CPU, HD, Net)
- Why?
  - concrete resources (e.g. 2 GHz CPU, 20 MB/s HD, 1 Gbit/s Net) unknown during component specification and implementation
- Thus: component developers provide SEFF specifications referring to resource types
- Once the concrete resource environment is specified, timing values can be derived

System Deployer

Resource Types in PCM

CPU

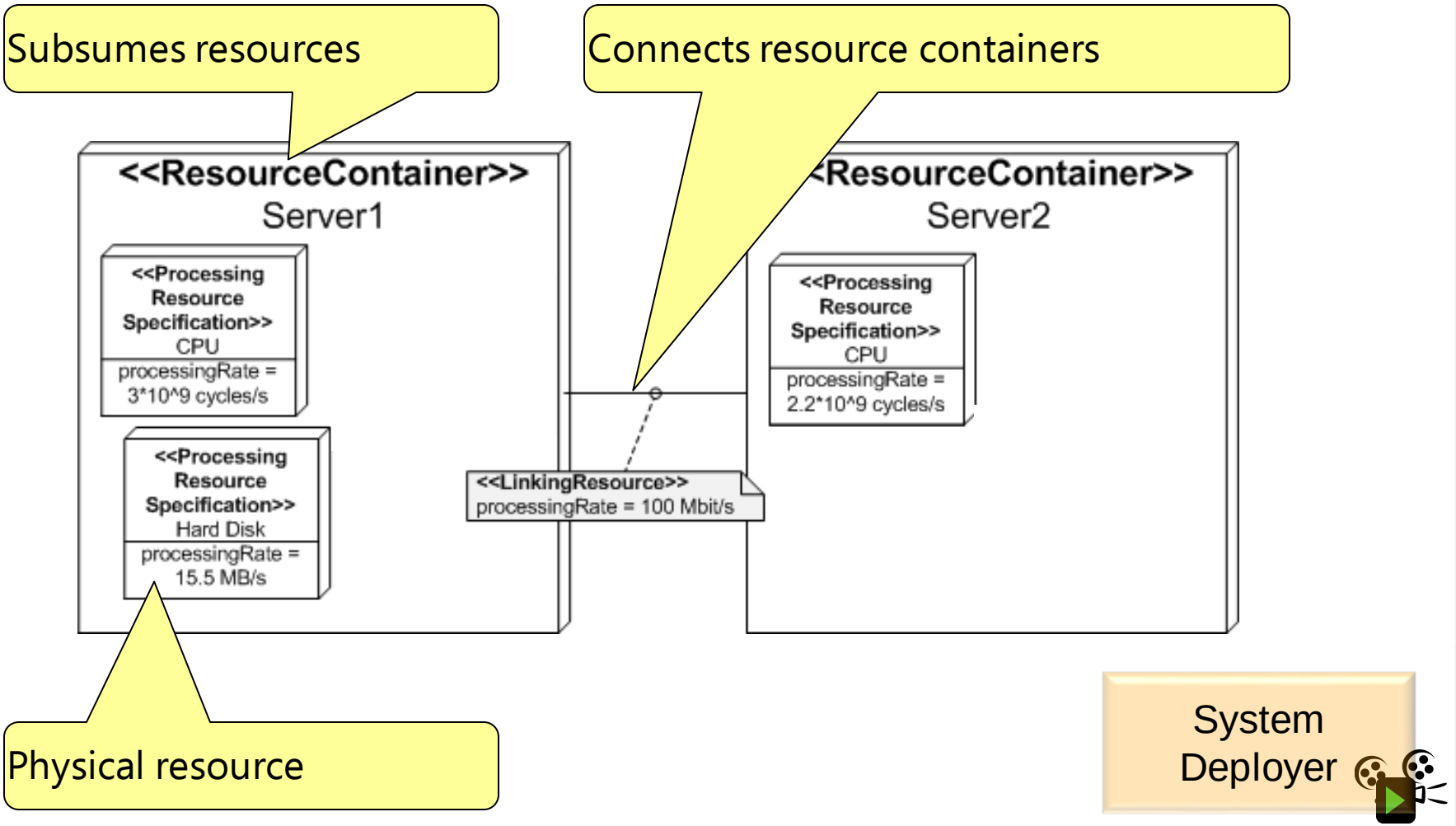
HD

Network

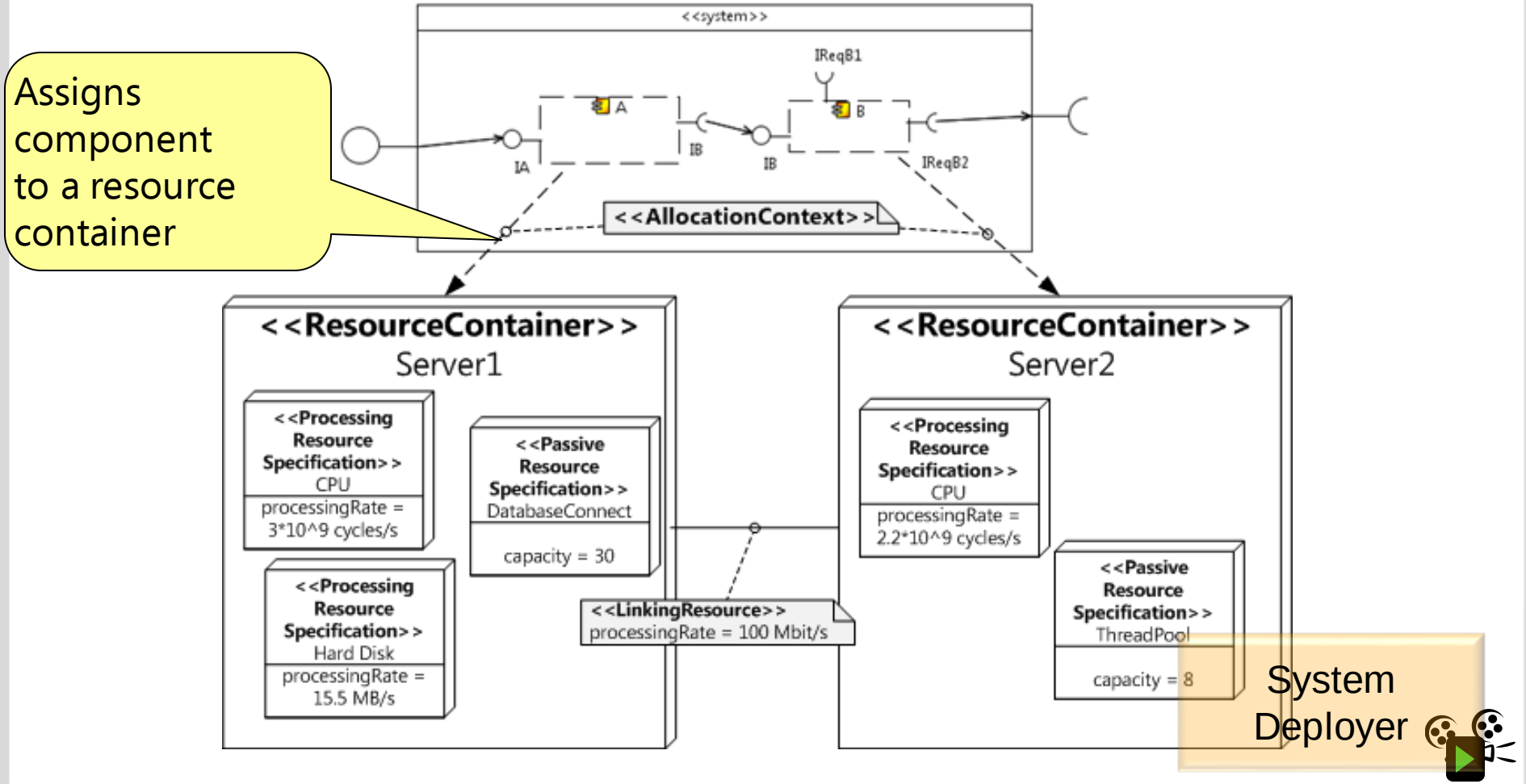
Memory

System Deployer

Example: Resource Environment in Palladio



Example: Allocation in Palladio



Software Architect vs. System Deployer

- Software Architect
  - Specification of a system
  - “Wiring” of components
- System Deployer
  - Resource types
  - Specification of a resource environment
  - Specification of an allocation

Overview

- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- **Roles in Component-based Software Development**
  - Component Developer
  - Software Architect
  - System Deployer
  - **Domain Expert**
- Conclusion / Summary

Domain Expert: Tasks

- Familiar with the business domain
- Specifies user behaviour
  - Number of users
  - User requests to the system
  - Input parameters characterisations

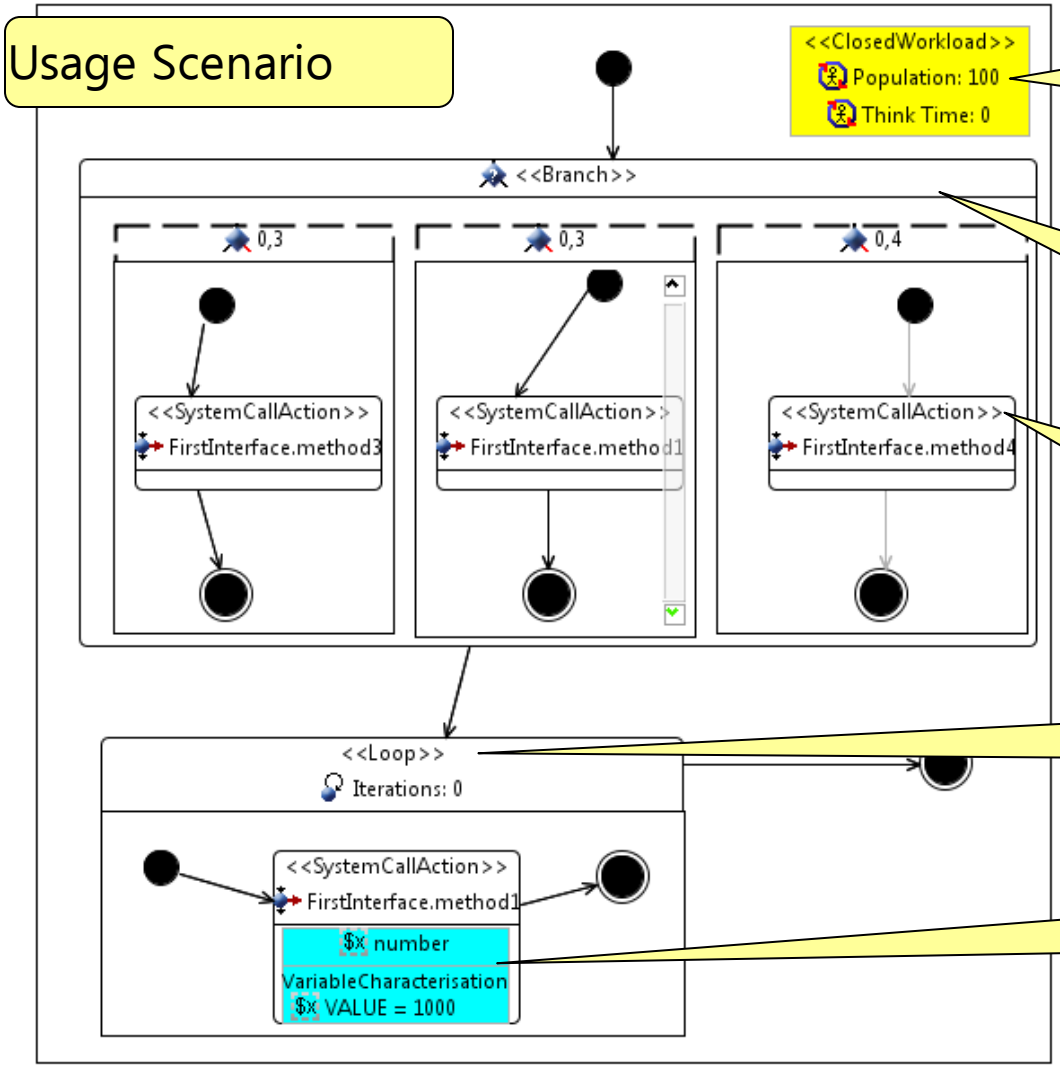
Domain Expert

Usage Model

- Models *user* behaviour, not component!
- Similar to SEFFs, but
  - Does not refer to resources
  - Does not refer to inner components of a system
  - Does not model parametric dependencies
  - Includes a workload specification
- Usage Model
  - *1…n* usage scenarios (1 per use case)
  - *1* workload per usage scenario

Domain Expert

Usage Model



Workload: open or closed

Branch

Call to a
system provided service

Loop

Variable Usage

Domain Expert

Overview

- Introduction
- Palladio
- Palladio Component Model (PCM) as a Modeling Language used by Palladio Approach
- Roles in Component -based Software Development
  - Component Developer
  - Software Architect
  - System Deployer
  - Domain Expert
- **Conclusion / Summary**

Summary: Contexts: Specified Properties

- Assembly Context
  - Horizontal Composition: Binding to other Components
  - Vertical Composition: Encapsulation in Composite Components
- Allocation Context
  - Allocation to Hardware Resources
  - Configuration
    - Component, Container
    - Communication
    - Security, Concurrency
- Usage Context
  - Usage at System Boundaries
  - User Arrival Rate
  - Number of Users
  - Request Probabilities
  - Parameter Values

[Becker2008]

Summary: Contexts and Roles

iObserve

Assembly Context	Allocation Context	Usage Context
Specified by Software Architect:	Specified by System Deployer:	Specified by Domain Expert:

▪ **Horizontal Composition: Binding to other Components**
▪ **Vertical Composition: Encapsulation in Composite Components**
	▪ **Allocation to Hardware Resources**
	▪ **Configuration**
	  ▪ Component, Container
	  ▪ Communication
	  ▪ Security, Concurrency
	  ▪ …
		▪ **Usage at System Boundaries**
		  ▪ User Arrival Rate
		  ▪ Number of Users
		  ▪ Request Probabilities
		  ▪ Parameter Values

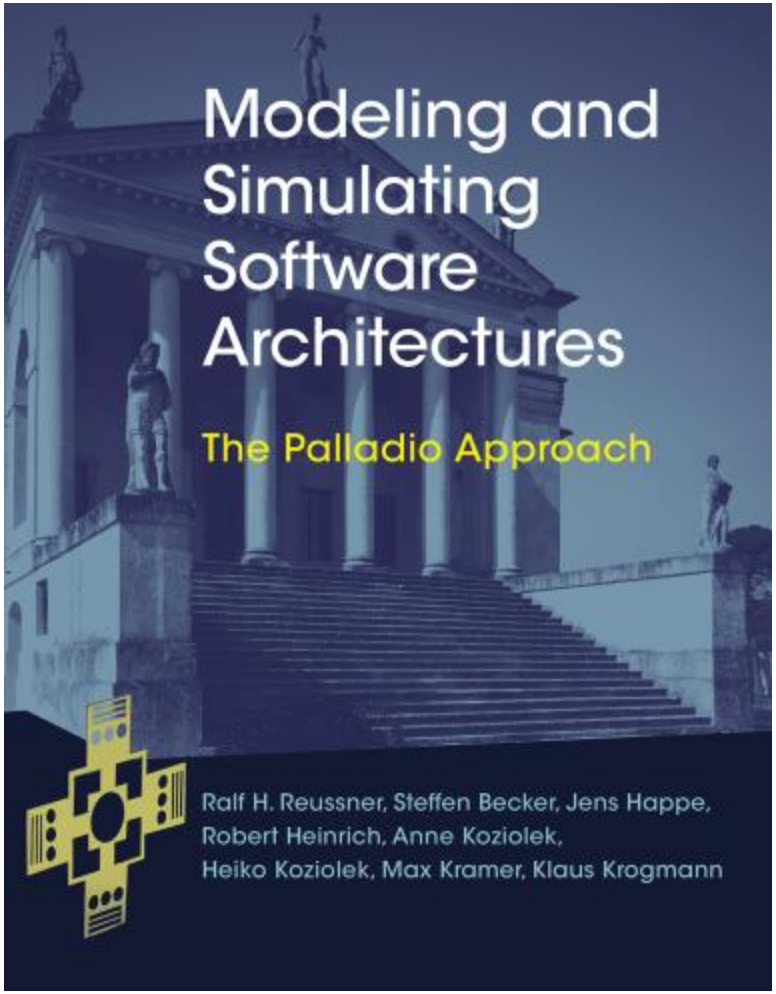
Computed by Tools:	Computed by Tools:	Computed by Tools:

▪ **Behaviour of the whole system**
  ▪ "Overall SEFF"
	▪ **Allocation-dependent QoS Characteristics**
	  ▪ Timing Values for Resource Demands
	  ▪ Failure Probabilities
	  ▪ …
		▪ **Usage inside Components**
		  ▪ Branch Probabilities
		  ▪ Loop Iteration Numbers
		  ▪ Input/Output Parameters
		  ▪ Usage-dependent Resource Demands

Palladio – Tool-Demo

- PCM Installation:
  - The PCM 4.0.0 release is only available for Eclipse 4.4 (Luna) and Eclipse 4.5 (Mars).
  - Download the Eclipse Luna Standard or Modeling Package:
    - http://www.eclipse.org/downloads/
  - Optional: Edit the eclipse.ini to change the memory settings to -Xms64m -Xmx2048m
  - Install Palladio from the site:
    - https://sdqweb.ipd.kit.edu/eclipse/palladiobench/releases/4.0.0/
- Media Store
  - https://svnserver.informatik.kit.edu/i43/svn/code/CaseStudies/MediaStore3/branches/PCM nightly build model/Model/MediaStore3_Model
  - Username: swaq2016
  - Password: swaq2016mediastore

Palladio Book



By Ralf H. Reussner, Steffen Becker, Jens Happe, Robert Heinrich, Anne Koziolek, Heiko Koziolek, Max Kramer and Klaus Krogmann

400 pp., 98 illus.
ISBN: 9780262034760

MIT Press, Cambridge, MA, October 2016.

References

- [Reussner 2016a]Ralf H. Reussner, Steffen Becker, Jens Happe, Robert Heinrich, Anne Koziolek, Heiko Koziolek, Max Kramer, and Klaus Krogmann Modeling and Simulating Software Architectures - The Palladio Approach, MIT Press 2016.
- [Becker2006a] Steffen Becker, Jens Happe, and Heiko Koziolek. Putting Components into Context: Supporting QoS-Predictions with an explicit Context Model. In *Proc. 11th International Workshop on Component Oriented Programming (WCOP'06)*, Ralf Reussner, Clemens Szyperski, and Wolfgang Weck, editors, July 2006, pages 1-6.
- [Becker2007a] Steffen Becker, Thomas Goldschmidt, Boris Gruschko, and Heiko Koziolek. A Process Model and Classification Scheme for Semi-Automatic Meta-Model Evolution. In *Proc. 1st Workshop MDD, SOA und IT-Management (MSI'07)*, 2007, pages 35-46. GiTO-Verlag. 2007.
- [Becker2008] Becker, Koziolek, Happe, Reussner, “Life-Cycle Aware Modelling of Component-based Software Architectures” CBSE2008