ICSA 2017 Tutorial
Runtime Modeling and Visualization
--
Introduction to Palladio
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00 – 09:10</td>
<td>Welcome and General Introduction</td>
</tr>
<tr>
<td>09:10 – 09:40</td>
<td>Study Foundations</td>
</tr>
<tr>
<td>09:40 – 10:00</td>
<td>Model-based Software Application Monitoring</td>
</tr>
<tr>
<td>10:00 – 10:30</td>
<td>Runtime Architecture Modeling and Visualization</td>
</tr>
<tr>
<td>10:30 – 11:00</td>
<td>Coffee Break</td>
</tr>
<tr>
<td>11:00 – 12:15</td>
<td>Introduction to the ExplorViz, Palladio, and iObserve Approaches with following Tool/Visualization Demos</td>
</tr>
<tr>
<td>12:15 – 12:30</td>
<td>Study Setup</td>
</tr>
<tr>
<td>12:30 – 14:00</td>
<td>Lunch</td>
</tr>
<tr>
<td>14:00 – 15:30</td>
<td>Comprehensibility Study</td>
</tr>
<tr>
<td>15:30 – 16:00</td>
<td>Coffee Break</td>
</tr>
<tr>
<td>16:00 – 16:30</td>
<td>Live Database Trace Visualization in Large Software Landscapes</td>
</tr>
<tr>
<td>16:30 – 17:00</td>
<td>Feedback and Open Discussion</td>
</tr>
</tbody>
</table>
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

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 (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
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

Robert Heinrich: Introduction to Palladio
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
- Analytical solvable
  - Performance jumps
  - Critical contention levels
- Simulation
  - Faster than "real" execution
Component Performance (1)
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
What are the intuitive inputs and outputs for a performance prediction model?

also see: http://www.youtube.com/watch?v=H0Gj-kdGhRs
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

Diagram:

- Person
  - 1..* assigned to
  - 1..* has role

- Role
  - 1 performed by
  - 1..* has task

- Task
Developer Roles

- Component Developers
- Software Architect
- System Deployer
- Domain Expert

[Becker2007a]

Robert Heinrich: Introduction to Palladio
Developer Roles

- Comp. Dev. DSL Instance
- Soft. Arch. DSL Instance
- Sys. Depl. DSL Instance
- Dom. Exp. DSL Instance
- Palladio Component Model
- Stochastic Regular Expr. Analysis
- SPA with Scheduling Analysis + Simulation
- Queueing Network Simulation
- Performance Prototype Execution + Measurement
- Java Code Skeletons Completion + Compilation

[Becker2007a] Robert Heinrich: Introduction to 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
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
Example: Component Description in Palladio

- Component interfaces need to be described
- Created components are stored in a repository
Example: Service Effect Specification: Idea

ComponentB

ComponentC

Component Developer

Robert Heinrich: Introduction to Palladio
Example: Service Effect Specification: Idea

**SEFF**

```
28ms a

a: b

b 20ms

28ms a

c 5ms
```

Component Developer

Robert Heinrich: Introduction to Palladio
Example: Service Effect Specification in Palladio

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

- Composed from Basic Components and/or other Composite Components
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: Tasks (2/2)

- Conducts **performance prediction** based on architectural specification

- **Delegates implementation tasks** to component developers

- **Guides** the whole development process
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
Example: System Composition in Palladio

- System is composed of components from repository
QoS Annotation

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

<<System>>

Execution Time = 250 ms

Software Architect
Example: Performance Evaluation in Palladio

[Image: A screenshot of SimuBench, showing a histogram with data points and a legend for Response Time of ConcurScenario.

Software Architect]
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
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
Resource Types in PCM

- CPU
- HD
- Network
- Memory
- System Deployer
Example: Resource Environment in Palladio

Subsumes resources

Connects resource containers

Physical resource

System Deployer
Example: Allocation in Palladio

Assigns component to a resource container.
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
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

- $1\ldots n$ usage scenarios (1 per use case)
- 1 workload per usage scenario
Usage Model

Usage Scenario

Workload: open or closed

Branch

Call to a system provided service

Loop

Variable Usage

Domain Expert

Robert Heinrich: Introduction to 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
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

<table>
<thead>
<tr>
<th>Assembly Context</th>
<th>Allocation Context</th>
<th>Usage Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specified by Software Architect:</td>
<td>Specified by System Deployer:</td>
<td>Specified by Domain Expert:</td>
</tr>
<tr>
<td>- <strong>Horizontal Composition:</strong> Binding to other Components</td>
<td>- <strong>Allocation to Hardware Resources</strong></td>
<td>- <strong>Usage at System Boundaries</strong></td>
</tr>
<tr>
<td>- <strong>Vertical Composition:</strong> Encapsulation in Composite Components</td>
<td>- <strong>Configuration</strong></td>
<td>- User Arrival Rate</td>
</tr>
<tr>
<td></td>
<td>- Component, Container</td>
<td>- Number of Users</td>
</tr>
<tr>
<td></td>
<td>- Communication</td>
<td>- Request Probabilities</td>
</tr>
<tr>
<td></td>
<td>- Security, Concurrency</td>
<td>- Parameter Values</td>
</tr>
<tr>
<td></td>
<td>- ...</td>
<td></td>
</tr>
<tr>
<td>Computed by Tools:</td>
<td>Computed by Tools:</td>
<td>Computed by Tools:</td>
</tr>
<tr>
<td>- <strong>Behaviour of the whole system</strong></td>
<td>- <strong>Allocation-dependent QoS Characteristics</strong></td>
<td>- <strong>Usage inside Components</strong></td>
</tr>
<tr>
<td></td>
<td>- “Overall SEFF”</td>
<td>- Branch Probabilities</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Loop Iteration Numbers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Input/Output Parameters</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Usage-dependent Resource Demands</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ...</td>
</tr>
</tbody>
</table>
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:
- 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/](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](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

References


