

Universität Karlsruhe (TH)
Forschungsuniversität gegründet 1825



Praktikum Ingenieurmäßige Software-Entwicklung

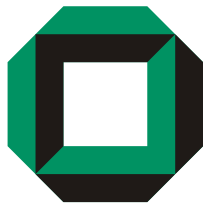
Palladio Component Model (PCM)

Prof. Dr. R. H. Reussner (reussner@ipd.uka.de)

Lehrstuhl Software-Entwurf und –Qualität

Institut für Programmstrukturen und Datenorganisation (IPD)

Fakultät für Informatik, Universität Karlsruhe (TH)



Outline



1. Introduction

- a. Roles, Process Model, Example
- b. Solver (Simulation, Analytical Model)

2. Component Developer

- a. Repository
- b. Component, Interface, Data Types
- c. SEFF

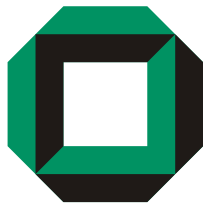
3. Stochastical Expressions

- a. Constants, PMF, PDF, Parameter Characterisation
- b. Parametric Dependencies

Lecture 1

Lecture 2

Lecture 3



Outline



4. Software Architect

- a) System (Composed Structure)
- b) QoS Annotations on System Interfaces

5. System Deployer

- a) Resource Types, Resource Environment
- b) Allocation

6. Domain Expert

- a. Usage Model
- b. Parameter Characterisations

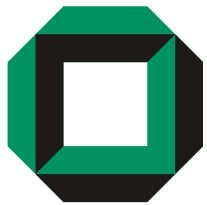
7. Solver, Result Interpretation

8. Comprehensive Case Study

9. Outlook

Lecture 4

Lecture 5



Outline



1. Introduction

- a. Roles, Process Model, Example
- b. Solver (Simulation, Analytical Model)

2. Component Developer

- a. Repository
- b. Component, Interface, Data Types
- c. SEFF

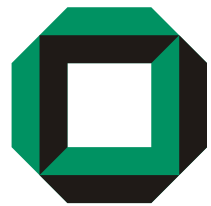
3. Stochastical Expressions

- a. Constants, PMF, PDF, Parameter Characterisation
- b. Parametric Dependencies

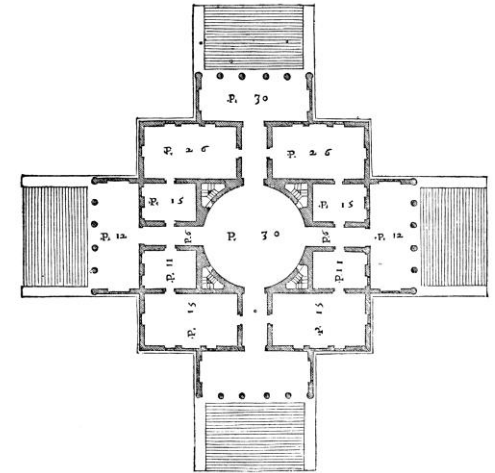
Lecture 1

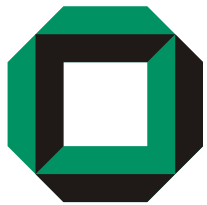
Lecture 2

Lecture 3

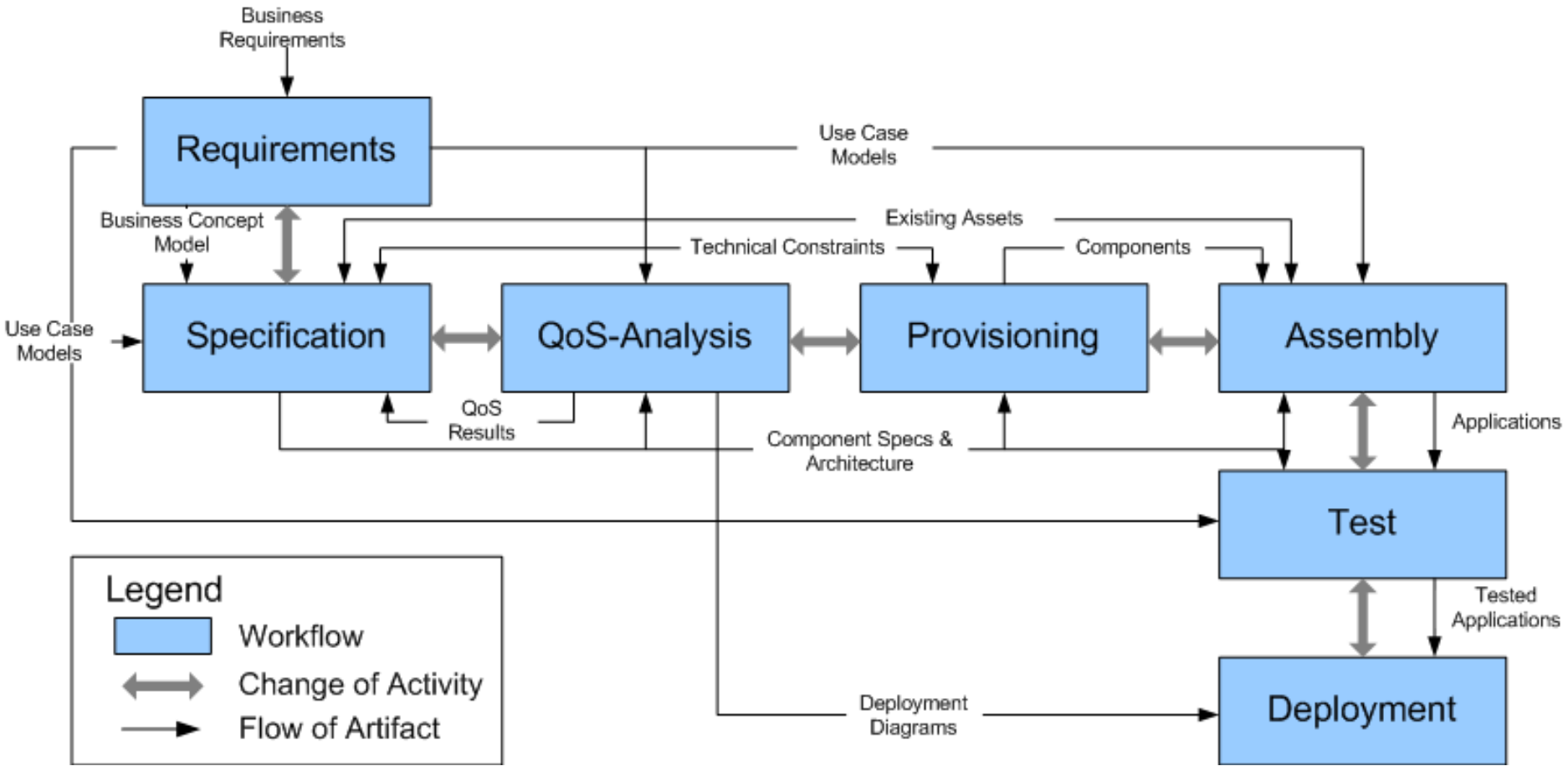


- Developed at Uni Oldenburg, Uni Karlsruhe since 2003
- Domain-specific Modelling Language
- Targeted at
 - Performance Prediction for Component-based Software Architectures
 - Business Information Systems
- Extensive Metamodel in EMF/Ecore
- Named after famous Renaissance Architect

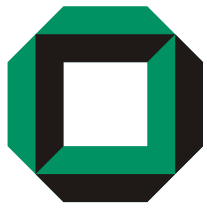




CBSE Development Process



[Cheeseman2000, Koziolk2006a]



Developer Roles

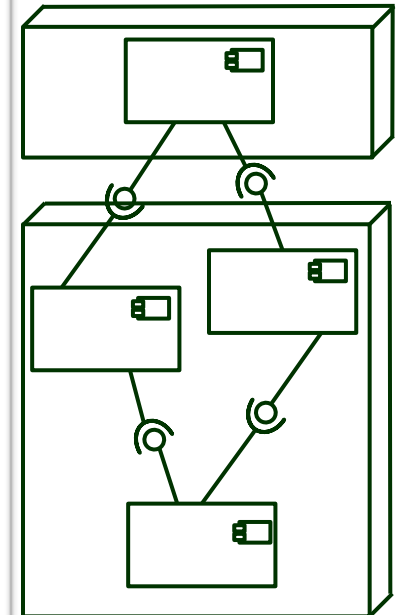
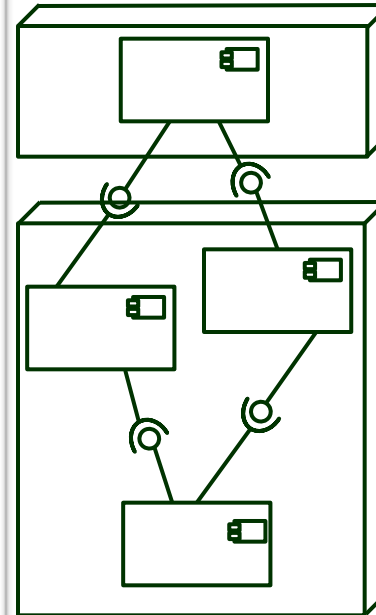
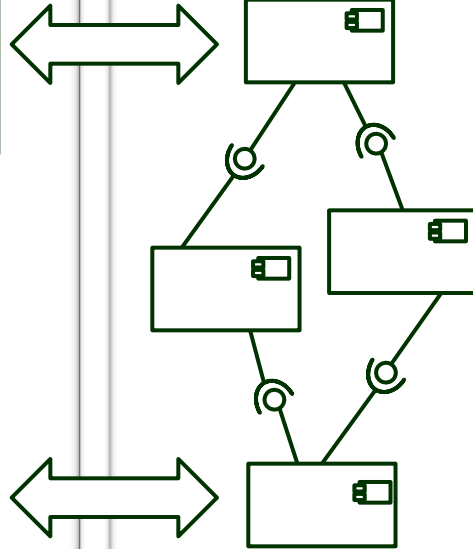
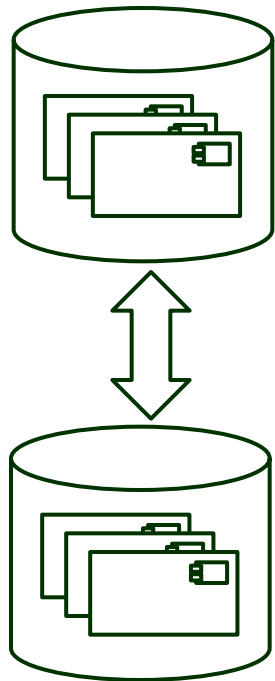


Component
Developers

Software
Architect

System
Deployer

Domain
Expert



[Becker2007a]

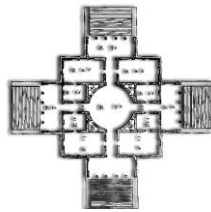
Comp.Dev.
DSL Instance

Soft. Arch.
DSL Instance

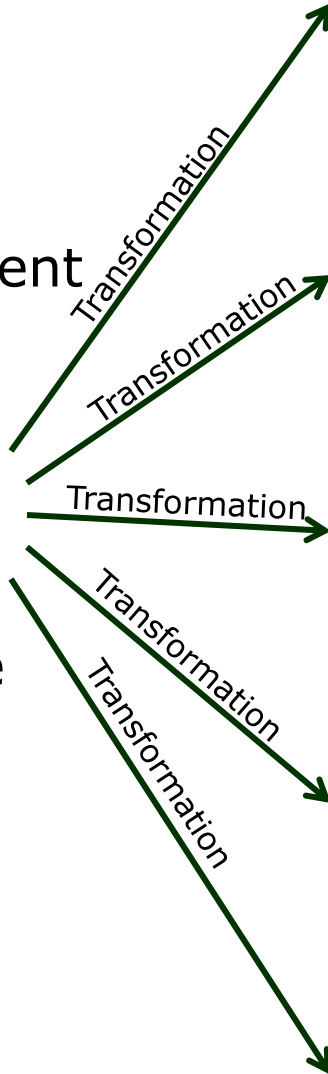
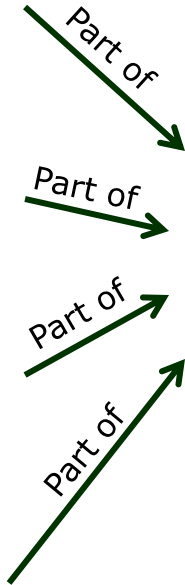
Sys. Depl.
DSL Instance

Dom. Exp.
DSL Instance

Palladio
Component
Model

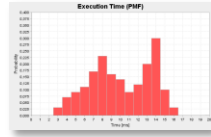


Instance



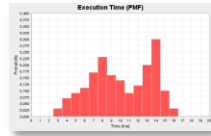
Stochastic
Regular Expr.

Analysis



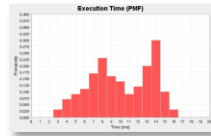
SPA with
Scheduling

Analysis +
Simulation



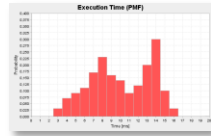
Queueing
Network

Simulation



Performance
Prototype

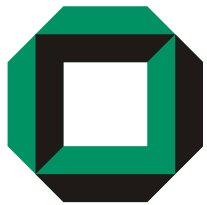
Execution +
Measurement



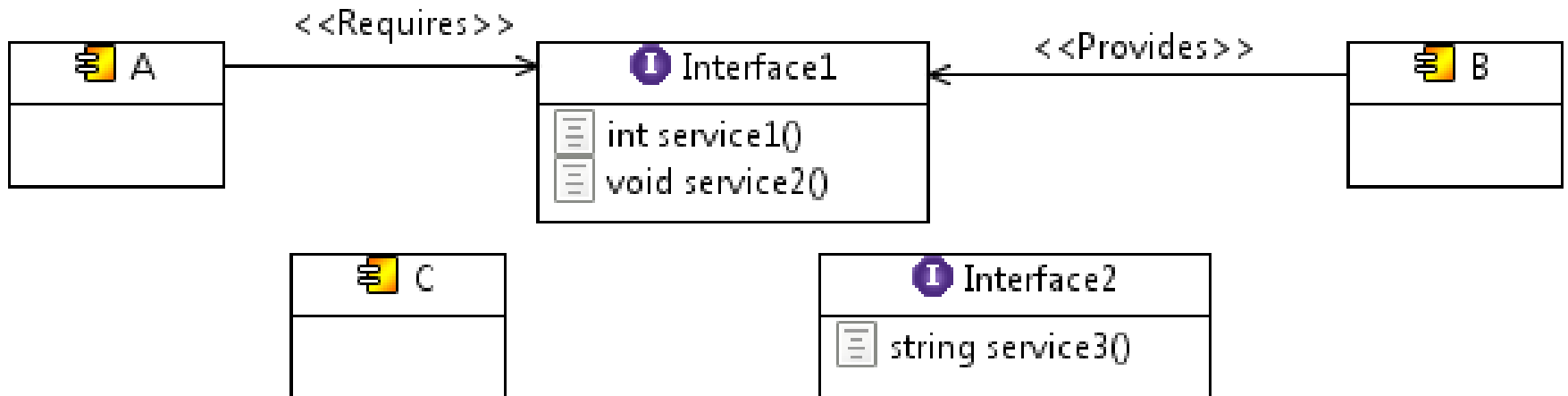
Java Code
Skeletons

Completion +
Compilation

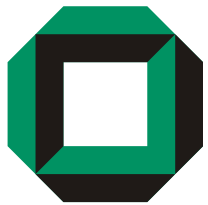
```
#include <nothing>
unsigned main()
{
  write : Hello all;
  write : I know !;
  write : not real;
  write : sp ;
  return all;
}
```



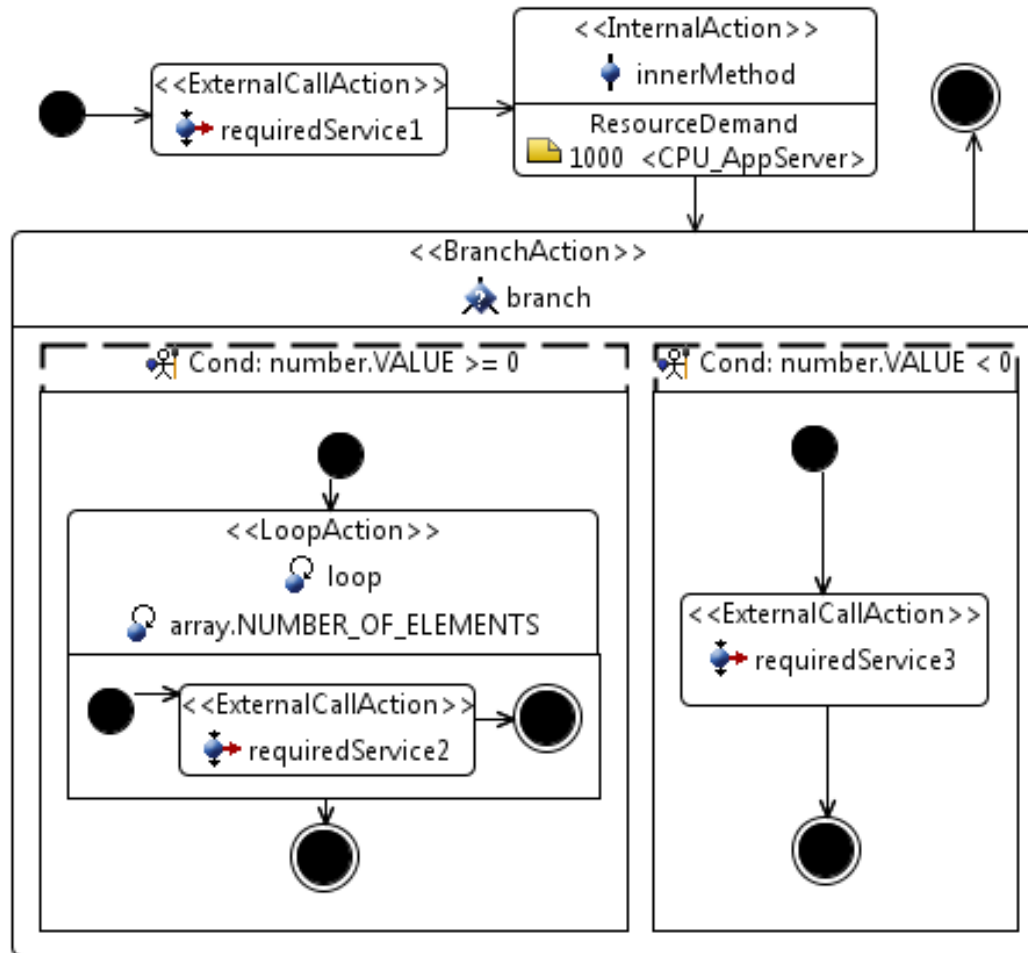
Repository



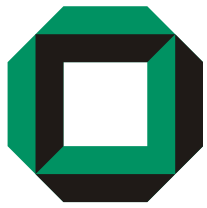
Component Developer



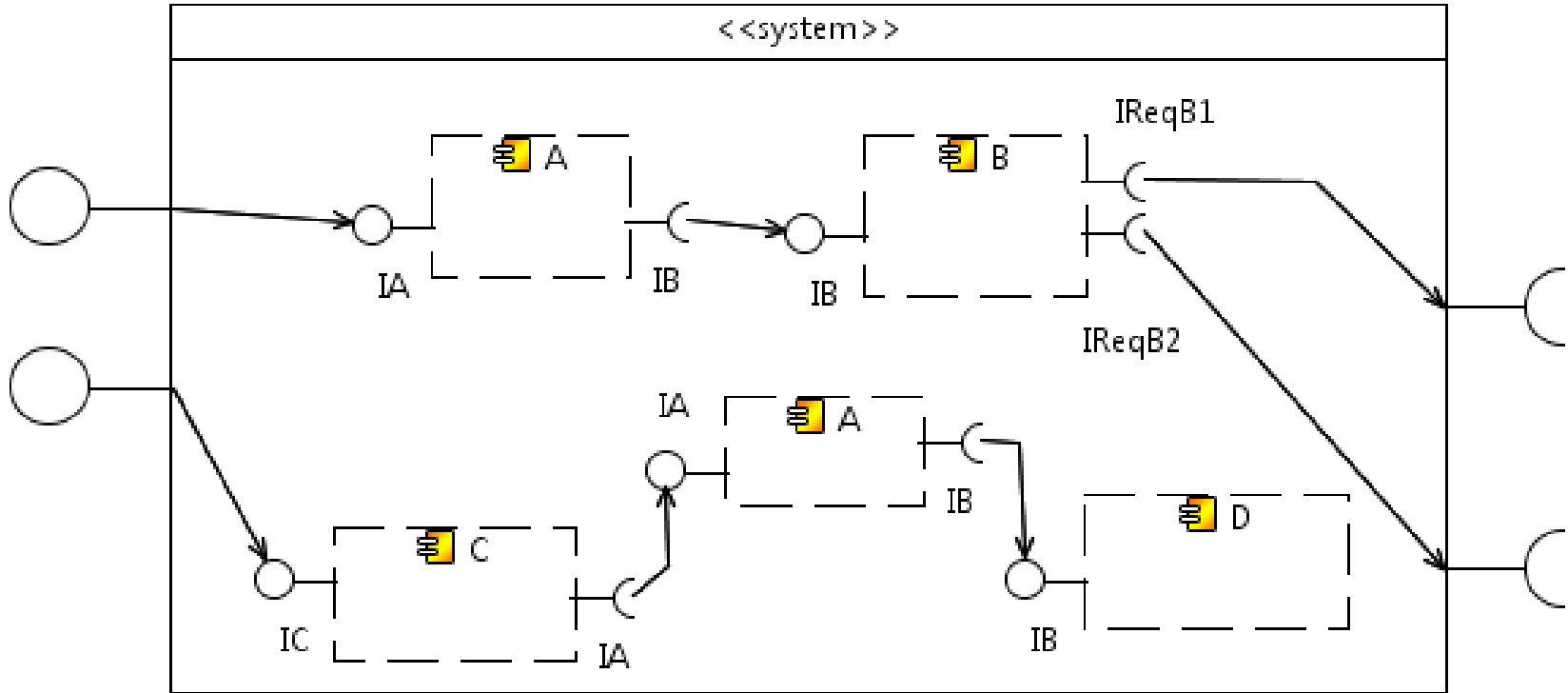
Service Effect Specification



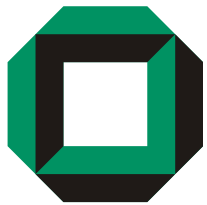
Component Developer



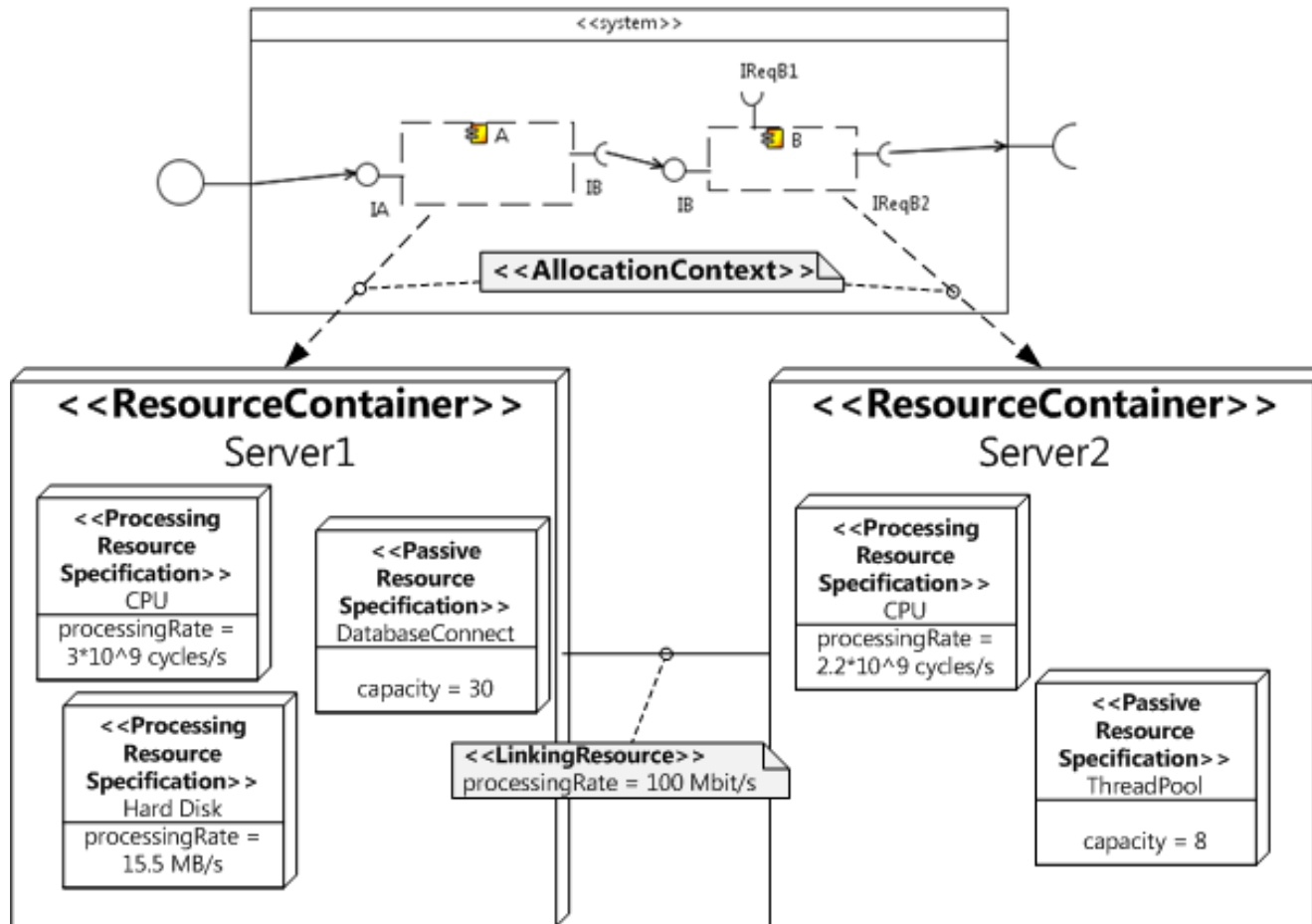
System



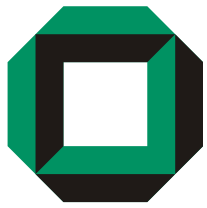
Software Architect



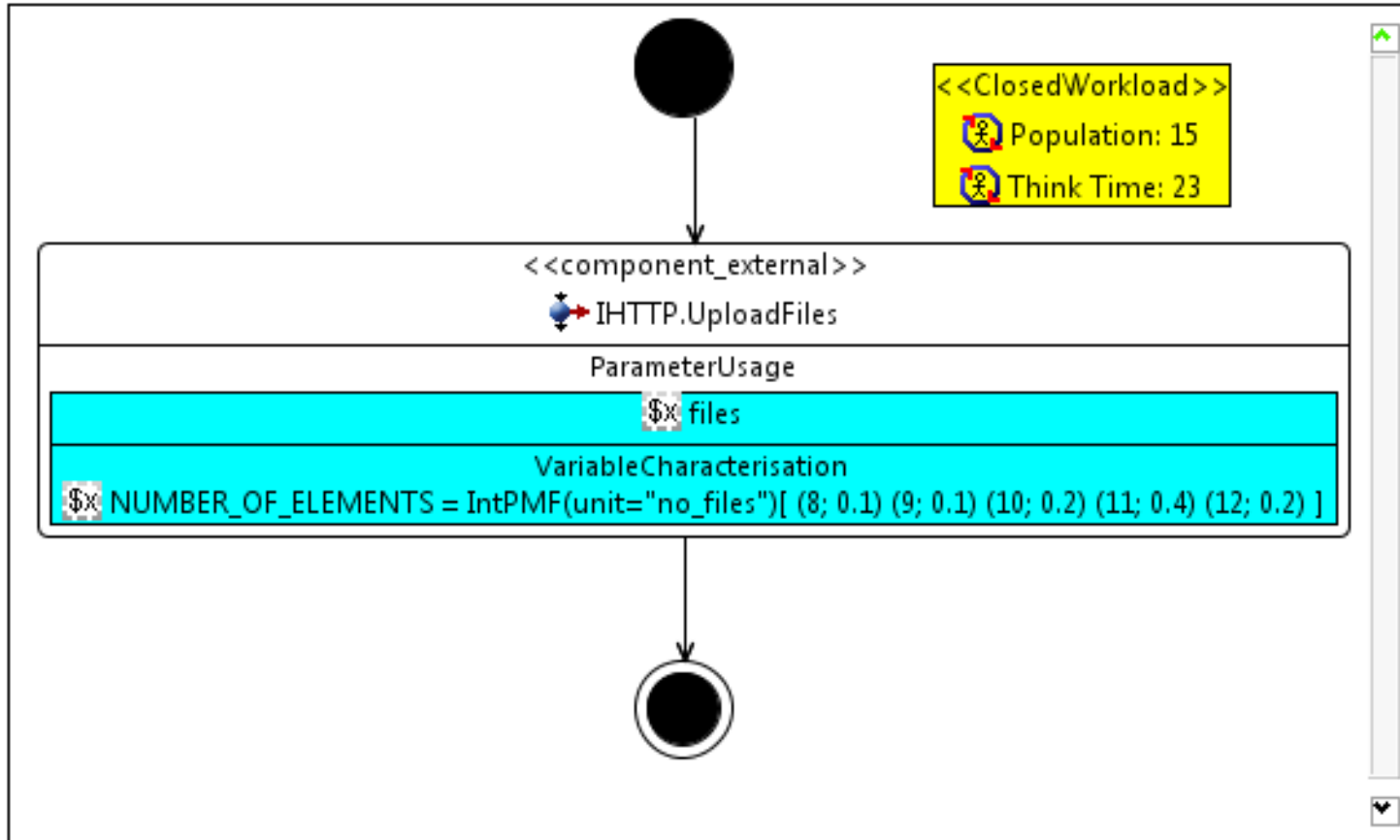
Resource Environment



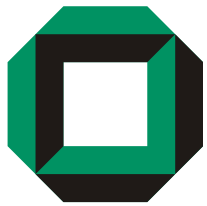
System
Deployer



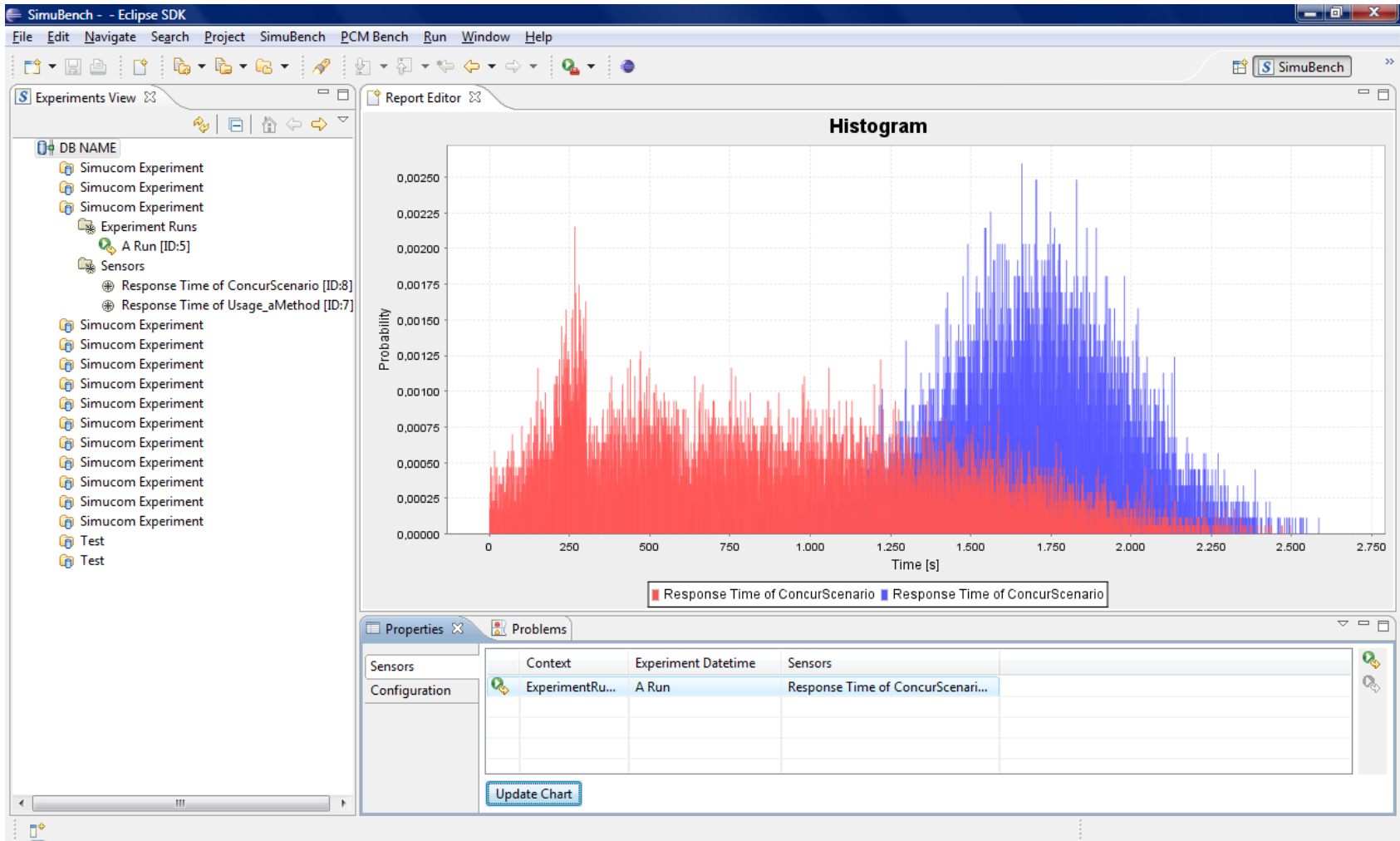
Usage Model

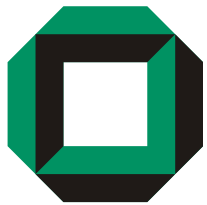


Domain Expert



Model Solver





Model Solver

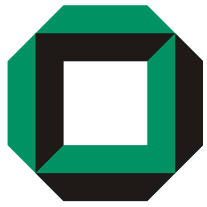


PCMSolver

- Only Single User
- Fast ($\sim 2-5$ seconds)
- Analytical Method, High Precision
- Stochastic Process Algebra based
- Traverses the architecture once
- Directly convolutes specified probability functions

SimuBench

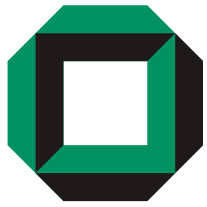
- Single + Multiple User
- Slow ($\sim 30-600$ sec.)
- Process-based Simulation
- Queueing Network based (G/G/n)
- Traverses the architecture repeatedly
- Draws samples from probability functions, adds them up



Example



- Blog-System 😊
- Switch to Eclipse!



Outline



1. Introduction

- a. Roles, Process Model, Example
- b. Solver (Simulation, Analytical Model)

2. Component Developer

- a. Repository
- b. Component, Interface, Data Types
- c. SEFF

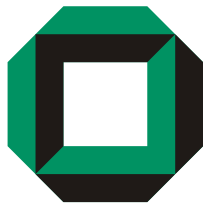
3. Stochastical Expressions

- a. Constants, PMF, PDF, Parameter Characterisation
- b. Parametric Dependencies

Lecture 1

Lecture 2

Lecture 3

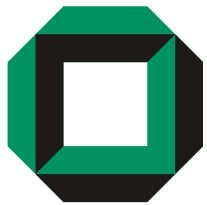


Tasks

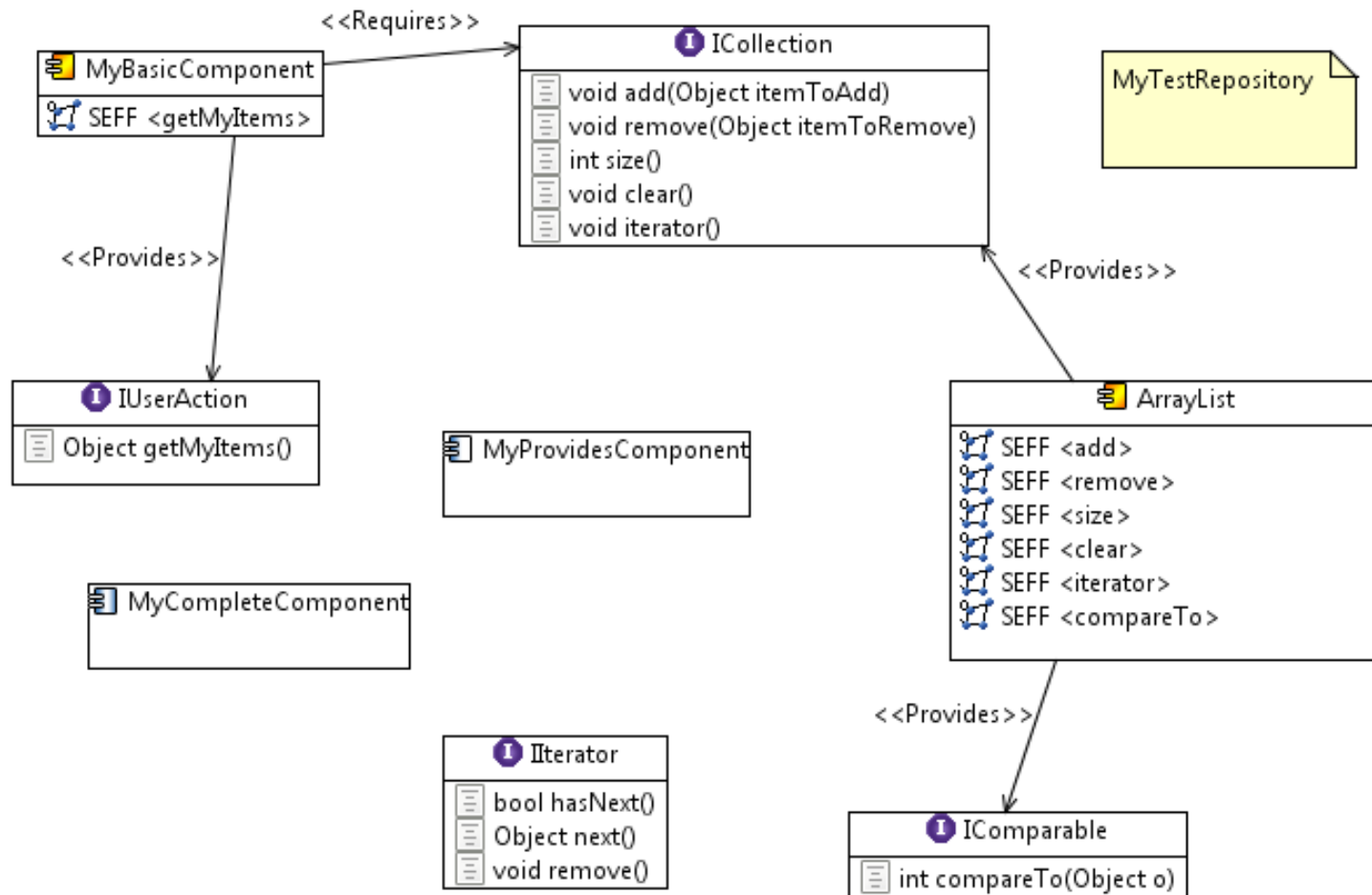


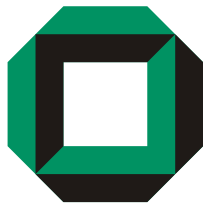
- Specifies Component & Interfaces
- Specifies Data Types
- Builds Composite Components
- Creates Service Effect Specifications
- Stores Modelling & Implementation Artefacts in Repositories
- Implements Components
- Tests Components
- Maintains Components

Component
Developer



Example Repository





Example Repository



ise.repository

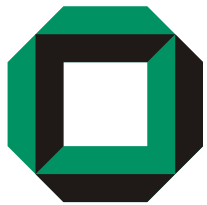
Resource Set

- platform:/resource/ISE2007/ise.repository
 - ISE2007 <Repository> [ID: _oy3rAACIEdyQbtXW6dE06A]
 - MyBasicComponent <BasicComponent> [ID: _Mg-30ACKEdyRt5tCnTJGxw]
 - Provided_IUserAction_MyBasicComponent <ProvidedRole> [ID: _Yw0cUACLEdyRt5tCnTJGxw]
 - Required_ICollection_MyBasicComponent <RequiredRole> [ID: _gpCyQACKEdyRt5tCnTJGxw]
 - Resource Demanding SEFF_Yw0cUQCLEdyRt5tCnTJGxw <ResourceDemandingSEFF> [ID: _Yw0cUQCLEdyRt5tCnTJGxw]
 - MyCompleteComponent <CompleteComponentType> [ID: _PO4ogACKEdyRt5tCnTJGxw]
 - MyProvidesComponent <ProvidesComponentType> [ID: _UcEfeACKEdyRt5tCnTJGxw]
 - ArrayList <BasicComponent> [ID: _aG6iAACKEdyRt5tCnTJGxw]
 - ICollection <Interface> [ID: _rc5BQACIEdyQbtXW6dE06A]
 - Signature add <Signature>
 - Signature remove <Signature>
 - Signature size <Signature>**
 - Signature clear <Signature>
 - Signature iterator <Signature>
 - IComparable <Interface> [ID: _jgeSYACKEdyRt5tCnTJGxw]
 - Signature compareTo <Signature>
 - Iterator <Interface> [ID: _B_xtcACLEdyRt5tCnTJGxw]
 - Signature hasNext <Signature>
 - Signature next <Signature>
 - Signature remove <Signature>
 - IUserAction <Interface> [ID: _TIgI8ACLEdyRt5tCnTJGxw]
 - Signature getMyItems <Signature>
 - Object <CompositeDataType> [ID: _O0nj1gCJEdyQbtXW6dE06A]

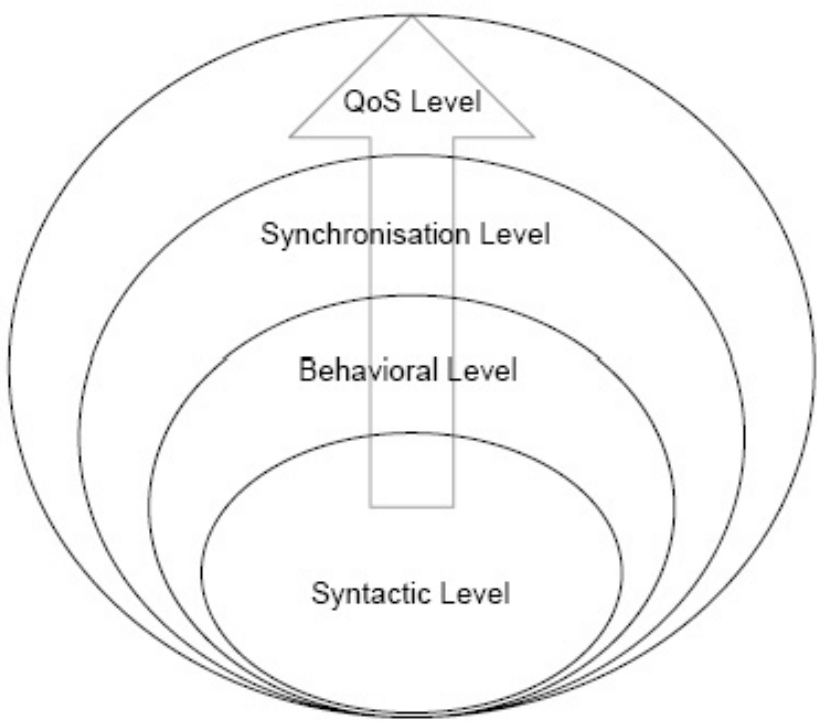
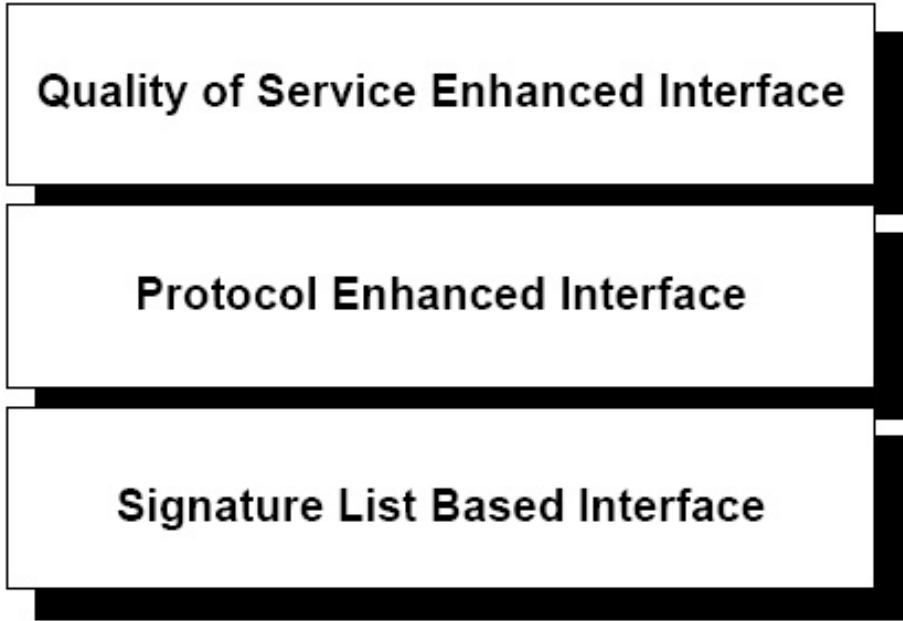
Selection Parent List Tree Table Tree with Columns

Problems Javadoc Declaration Console Error Log Progress Properties

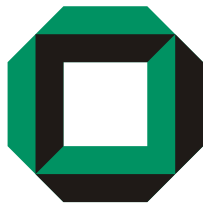
Property	Value
Returntype Signature	INT <PrimitiveDataType>
Service Name	size



Interfaces



[Beugnard1999]

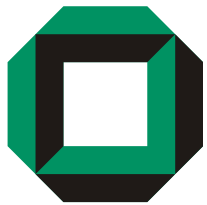


PCM Interfaces

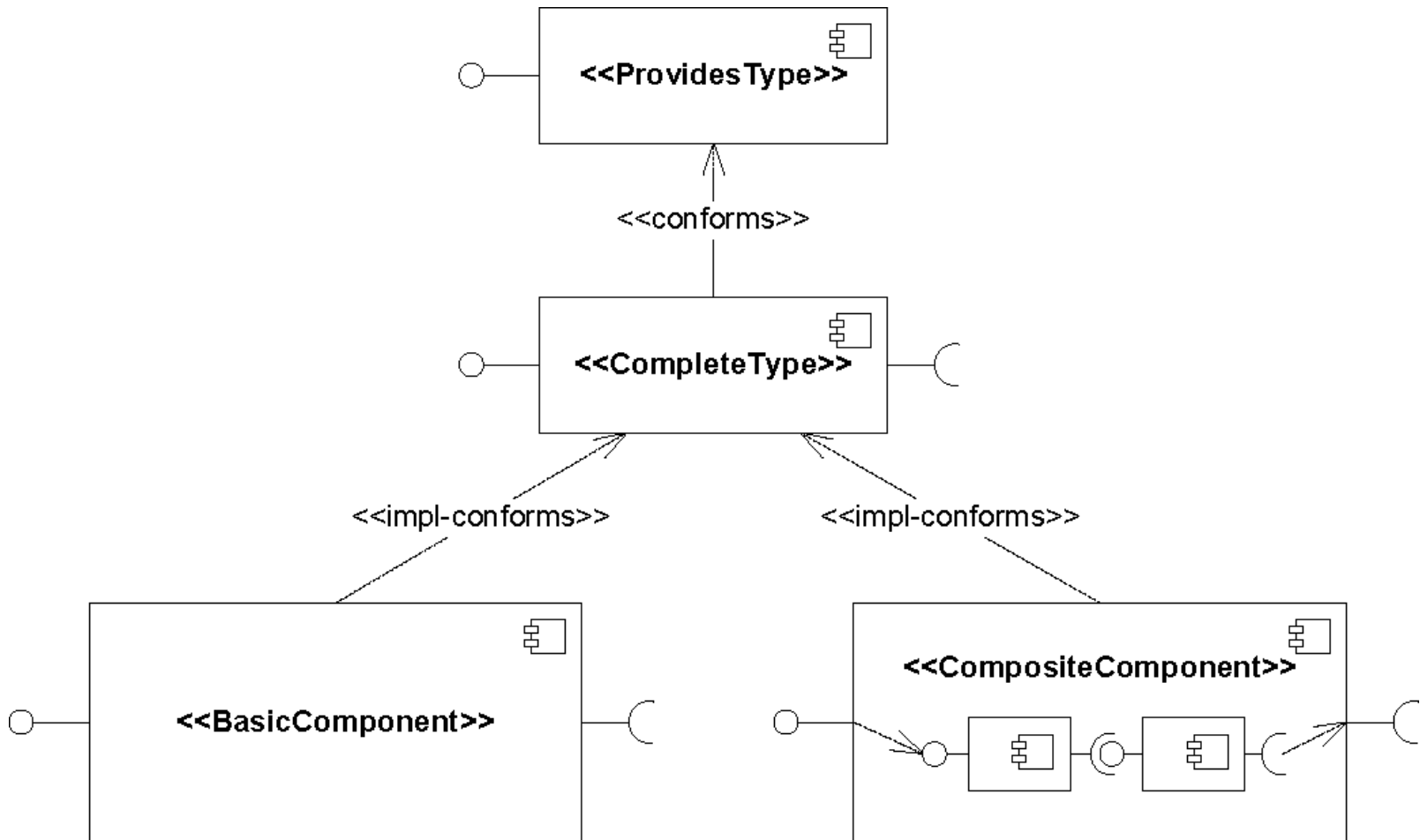


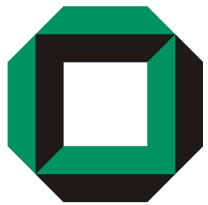
- QoS (=Performance, Reliability)
 - Service Effect Specification (Lecture 2)
- Protocol (=Valid Call Sequences)
 - Finite State Machine (Not shown here)
- Signature
 - Corba IDL:
 - Return Type
 - Name
 - Parameter List
 - Exception List

I Collection	
	void add(Object itemToAdd)
	void remove(Object itemToRemove)
	int size()
	void clear()
	void iterator()



PCM Component Types



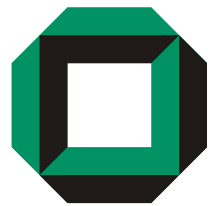


Provided Component Type



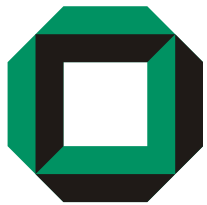
- Only Provided Interfaces mandatory
- May contain required services, not mandatory
- Specified during early development, refined later
- Situation: certain functionality needed, but additionally required services unknown
- QoS Annotations





- Provided and Required Interfaces mandatory
- Dependencies between Provided and Required Interfaces not fixed
- Situation: Needed functionality known, component environment also fixed



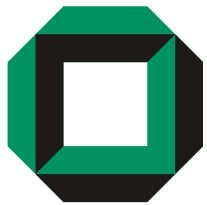


Basic Component



- Provided/Required Interfaces mandatory
- One option to implement a Complete Type
- Service Effect Specification for Dependencies between Provided and Required Interfaces
- May be composed to Composed Components

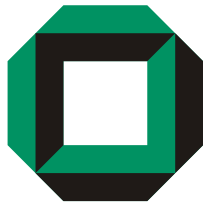




Composite Component



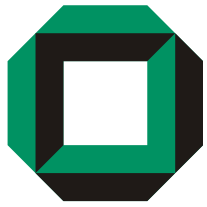
- During development composed from any component types
- Finally composed from Basic Components and/or other Composite Components
- Likely not used in the experiment, but may occur in exercises



Data Types



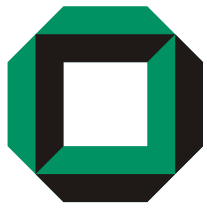
- Primitive Datatype
 - INT, CHAR, BOOL, DOUBLE, LONG, ...
- Collection Datatype
 - Contains an inner primitive datatype
 - ARRAY, SET, LIST, TREE, HASHMAP, ...
- Composite Datatype (Struct)
 - Contains inner primitive and/or collection and/or composite datatypes
 - ADDRESS, CUSTOMER, PERSON, ...



Hands on Example



- Switch to PCMBench



Outline



1. Introduction

- a. Roles, Process Model, Example
- b. Solver (Simulation, Analytical Model)

2. Component Developer

- a. Repository
- b. Component, Interface, Data Types
- c. SEFF

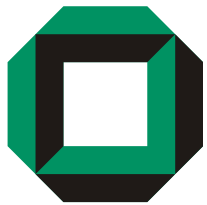
3. StoEx

- a. Constants, PMF, PDF, Parameter Characterisation
- b. Parametric Dependencies

Lecture 1

Lecture 2

Lecture 3



Lessons Learned Today



- Person – Role – Task
- Component Developer, Software Architect, System Deployer, Domain Expert
- PCMSolver vs. SimuBench
- PCM Repository (Component Developer)
 - Components
(Provided, Complete, Basic, Composite)
 - Interfaces
(Signature, Protocol, SEFF)
 - Data Types
(Primitive, Collection, Composite)