

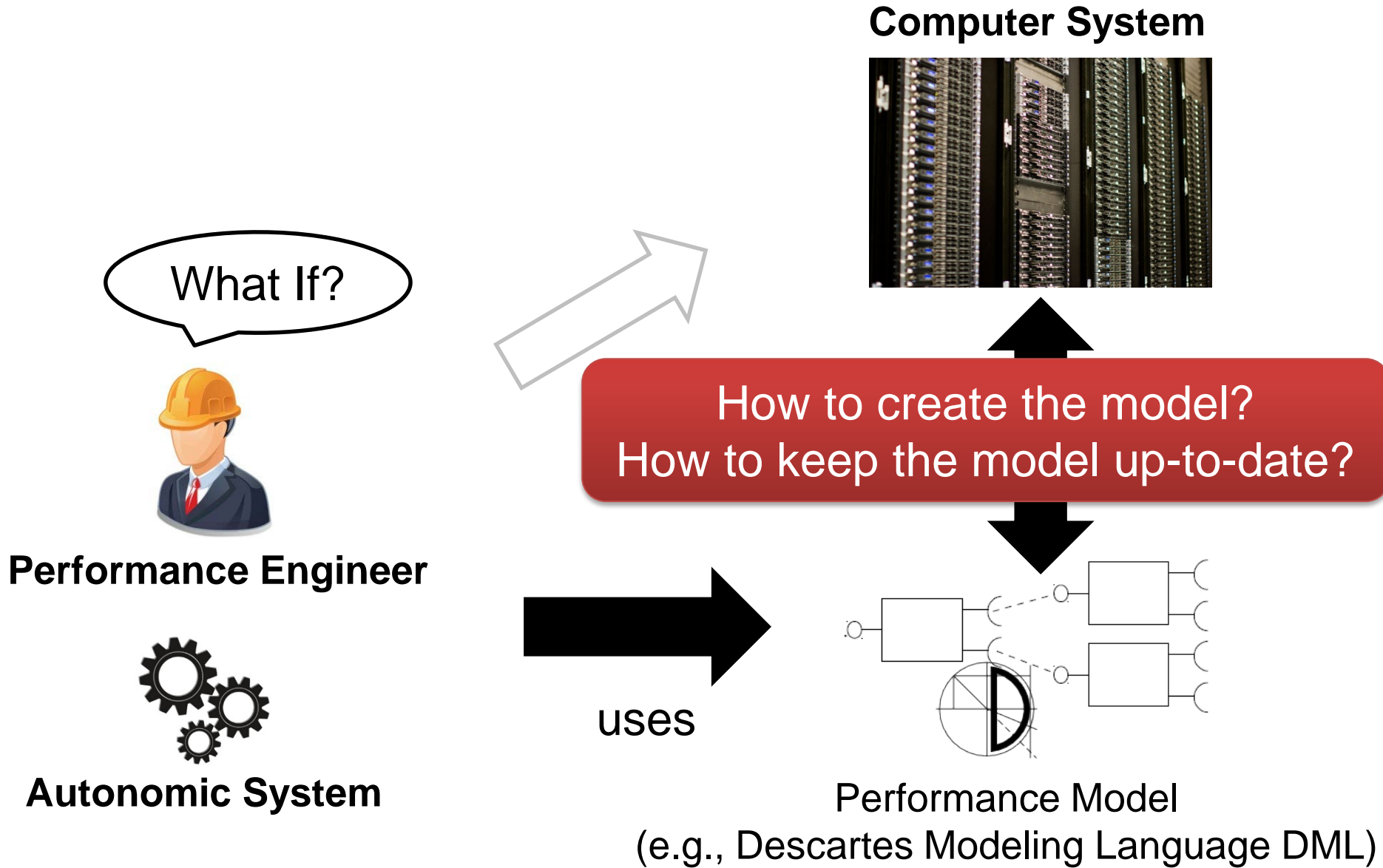
A Reference Architecture for Online Performance Model Extraction in Virtualized Environments

Simon Spinner, Jürgen Walter, Samuel Kounev

University of Würzburg – Chair of Software-Engineering
<http://se.informatik.uni-wuerzburg.de/>

March 12, 2016

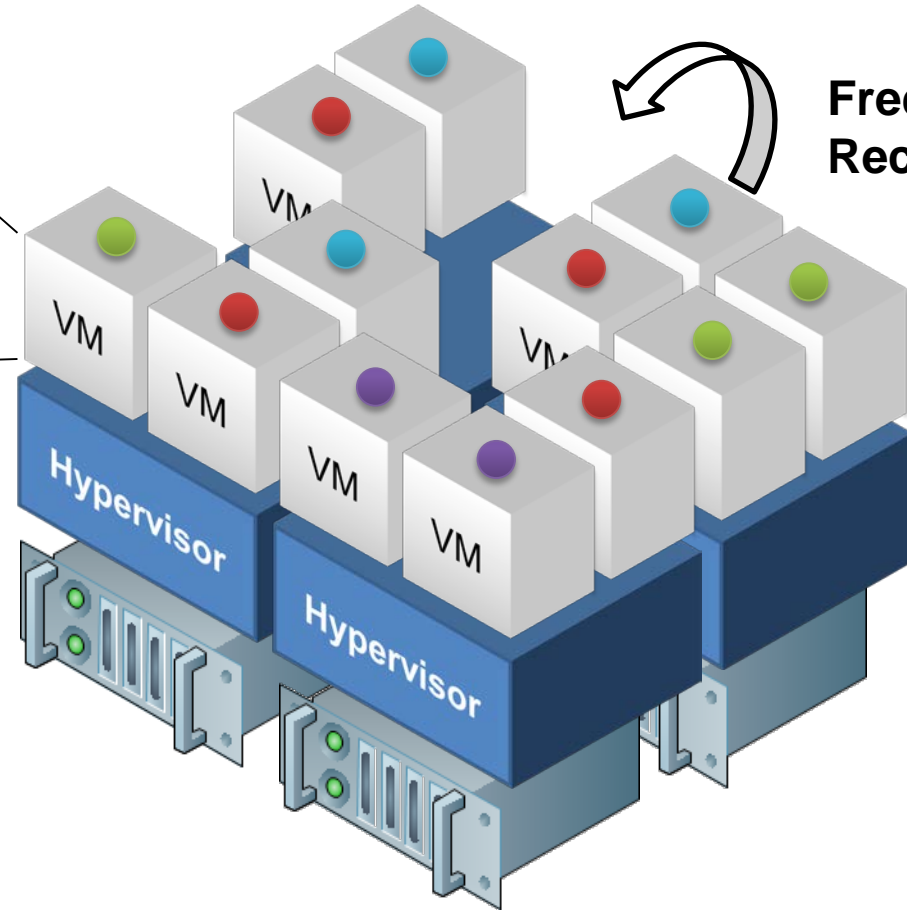
WOSP-C 2016, Delft, the Netherlands



Heterogeneous Technologies ● ● ● ●

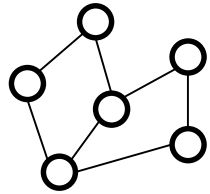


Complex Technology Stacks



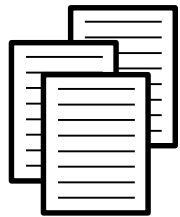
Frequent Reconfigurations

Performance Model Extraction



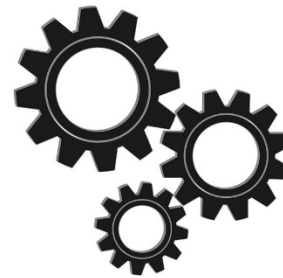
Architecture Knowledge

- Components
- Servers
- Control flow



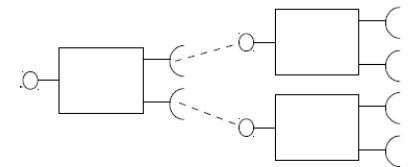
Observation Data

- System-level
- Application-level



Model Extraction

- Model structure
- Model parameters



Performance Model

- Architecture-level

Technology-specific



Technology-agnostic



Related Work

- (Layered) Queueing Networks

 - Hrischuk et al. (1999)

 - Awad and Menascé (2014)

generic

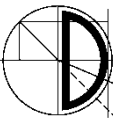
- Palladio Component Model (PCM)

 - Krogmann (2010)

 - Brosig et al. (2011)

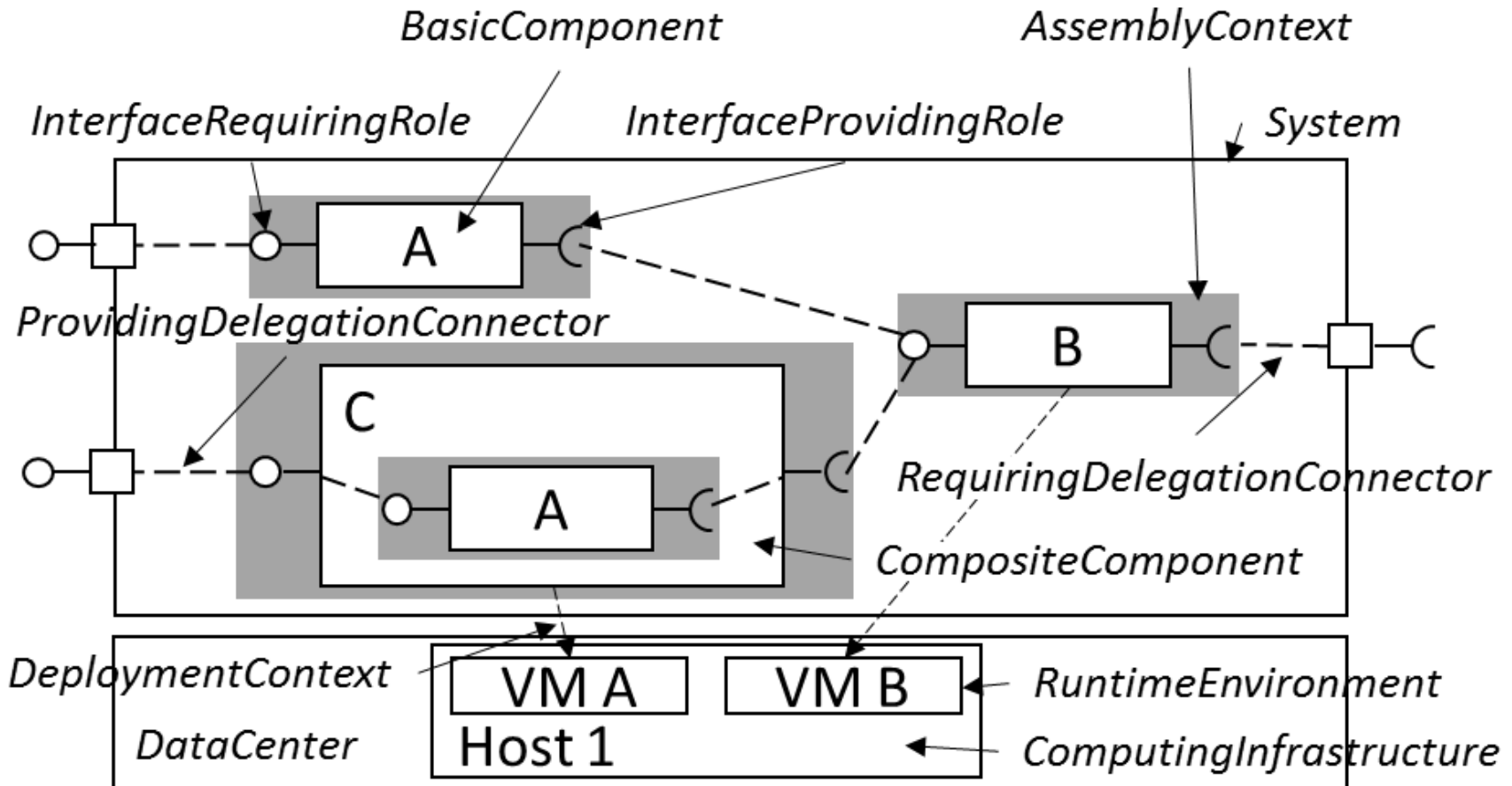
 - Brunnert et al. (2013)

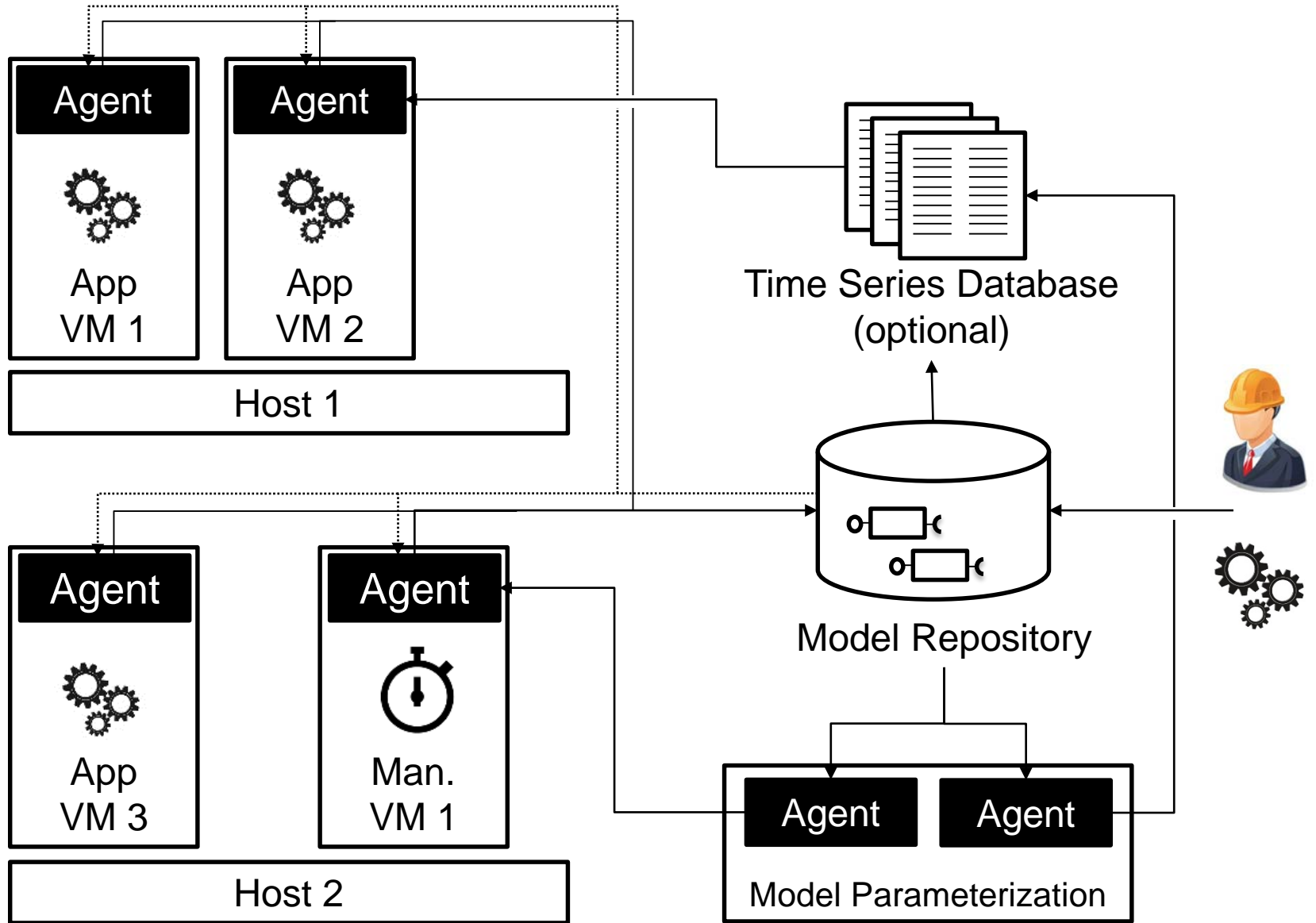
J2EE specific



- „Performance-aware“ virtual appliances
 - Pre-packaged technology stack
 - Self-reflection through a special agent
- Agent provides
 - Model skeleton
 - Monitoring data
- A model skeleton is a submodel, which is
 - Unparameterized
 - Composable

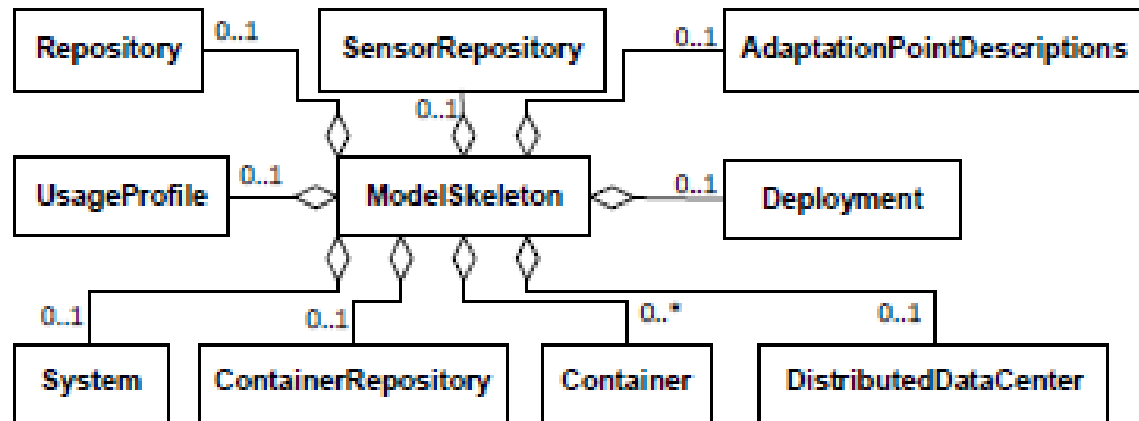
Descartes Modeling Language (DML)





Model Skeleton Agent

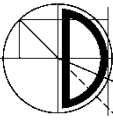
- Extracts model structure, e.g.:
 - Components
 - Component assembly / control flow
 - Resource landscape
- Represents local view of agent
- MOF-based metamodel



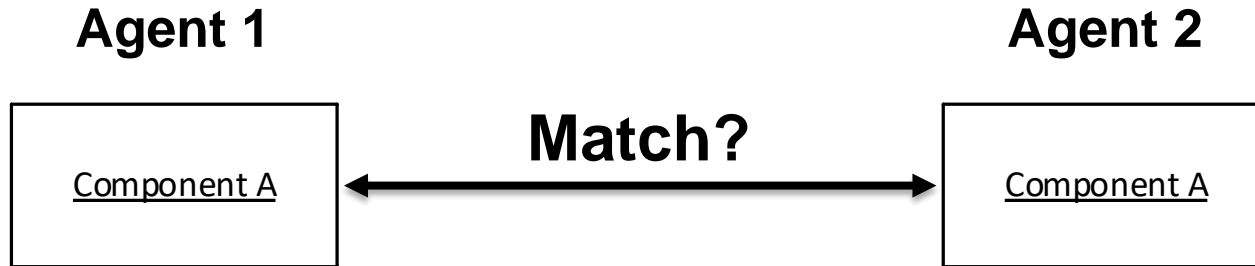
Monitoring Agents

- Provides:
 - Description of available sensors
 - Current monitoring data
- Examples:
 - System-level: CPU utilization of virtual machine
 - Application-level:
 - Response-time
 - Throughput

Model Parameterization Agent



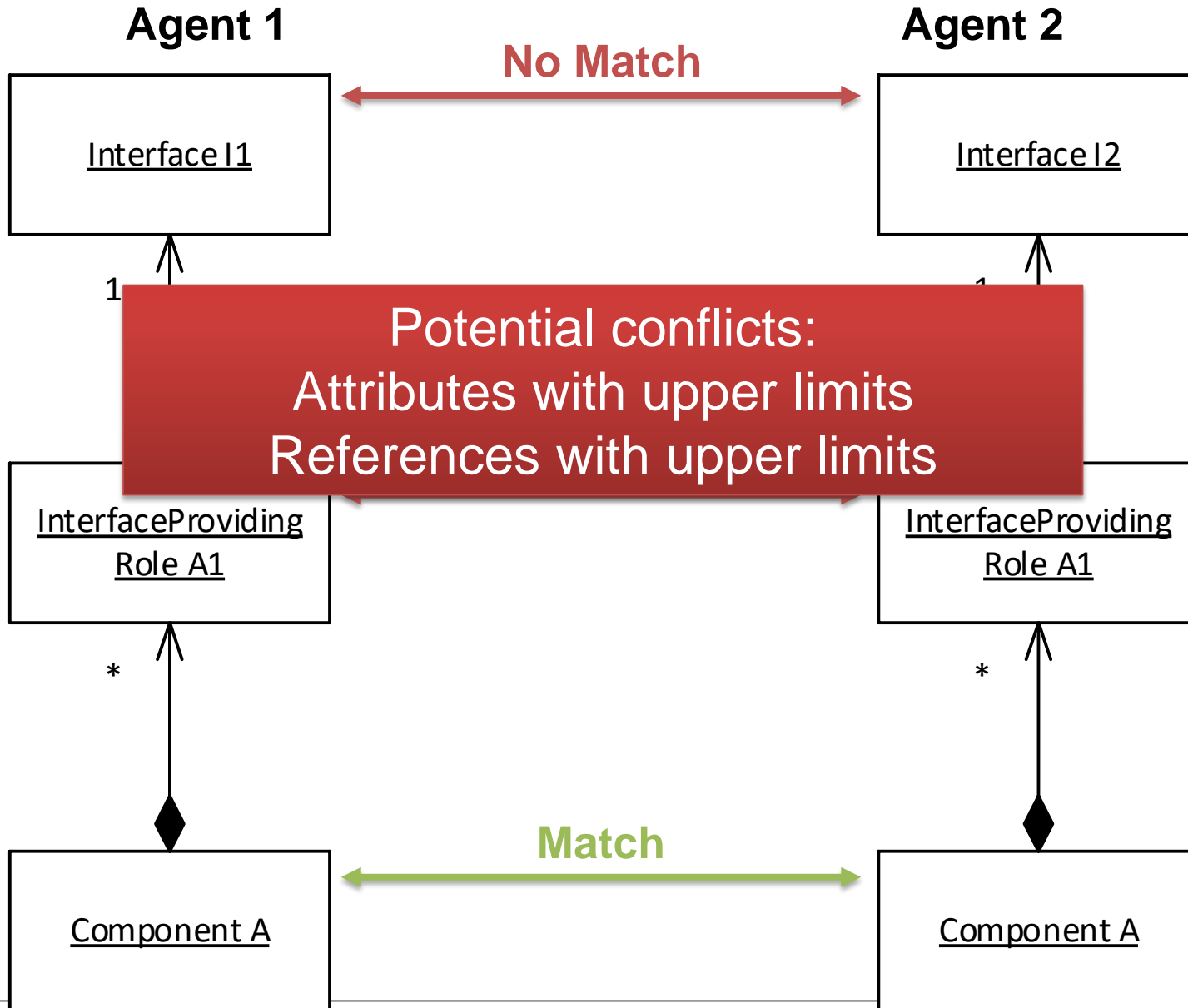
- Uses information provided by
 - Model structure agents
 - Monitoring agents
- Calculates parameter values for a given time
- Examples
 - Resource demands
 - Branching probabilities
 - Load intensity



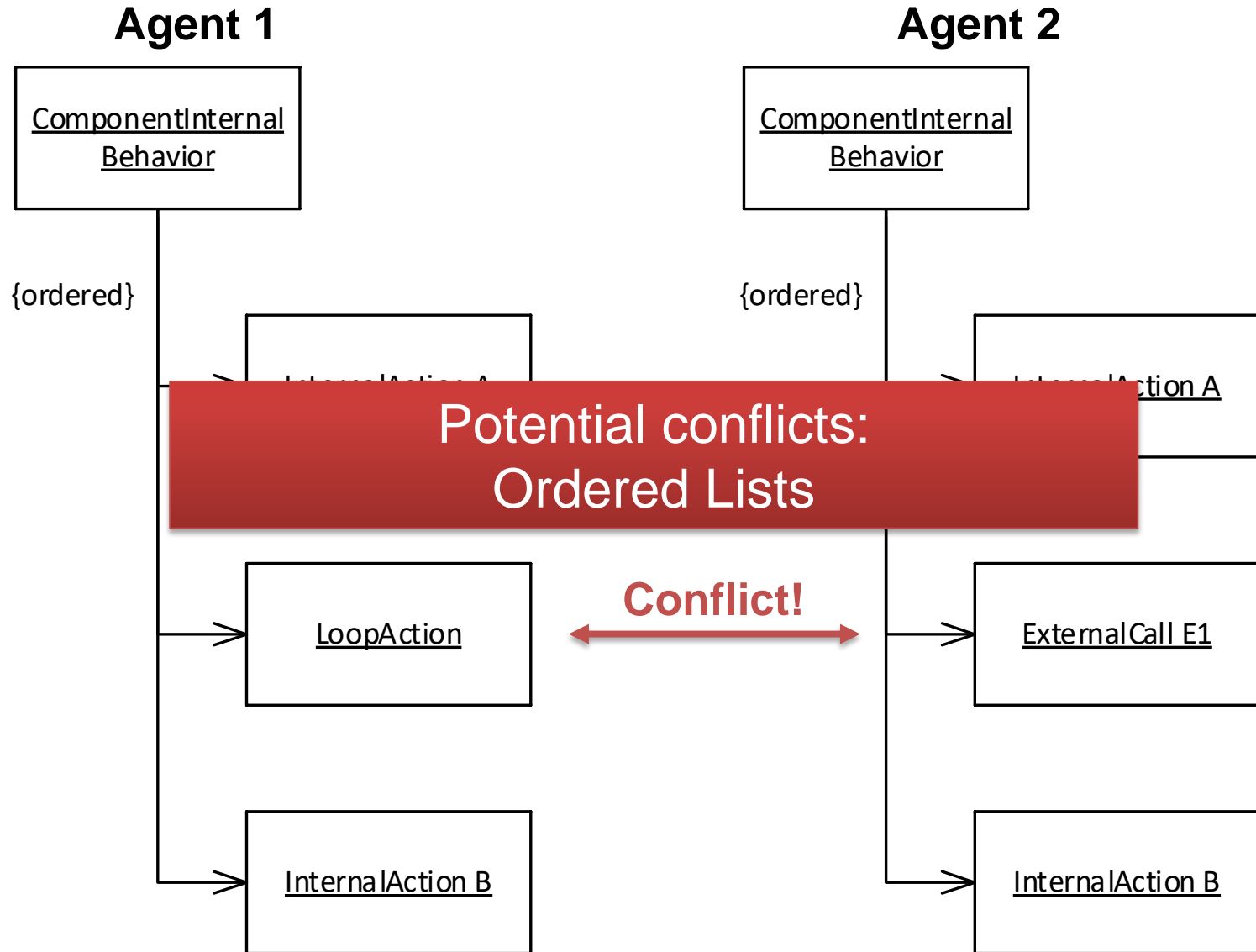
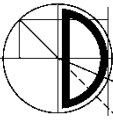
- Agents are independent
 - Randomly generated identifiers cannot be used
 - Unique technical identifiers required
- Types of model elements
 - Instance-level (e.g., virtual machine)
 - Type-level (e.g., component level)

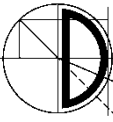
Merging required!

Merge Example 1

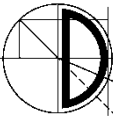


Merge Example 2





- Model Repository
 - EMF-based
 - Eclipse CDO
- Agents
 - VMware vSphere (through web service API)
 - Wildfly (former JBoss) extension
 - LibReDE



- „Performance-aware“ virtual appliances
 - Reflective capabilities provided by agents
 - Agents are tailored for certain technology stacks
 - Provide model skeletons of their local view
- Model repository
 - Merges model skeletons
 - Creates global performance model
 - Determines parameter values for model variables

Thank you!

Simon Spinner, Jürgen Walter, Samuel Kounev

University of Würzburg – Chair of Software-
Engineering

<http://se.informatik.uni-wuerzburg.de/>