Service Orchestration @ FIA Session
“Orchestration Across Networks, Services, Things and Content”

Dr. Dimka Karastoyanova
Junior Professor
Dimka.Karastoyanova@informatik.uni-stuttgart.de
Institute of Architecture of Application Systems (IAAS)
University of Stuttgart, Germany

University of Stuttgart
Universitätsstr. 38, 70569
Stuttgart
Germany

Phone +49-711-7816 476
Fax +49-711-7816 472
Office 1.361
Topics

- Services and service compositions
- Infrastructures
- Challenges / research topics
  - Service based application (SBAs)
  - SBA and Service-based Infrastructures
Services and Service Orchestration
SOA, Services

- Service Oriented Architecture
  - An architectural style for application integration within and across organisational boundaries
  - Basic concept: services
  - Basic model: service client, service provider, discovery component
  - Operations: Publish, Find, Bind/invoke/interact
  - Enables **virtualization** of applications
  - Independent of implementation and platform
  - **Loose coupling** – Messaging and stable standard interfaces
  - Integration of application systems
Service Composition

- **Services** can be composed into
  - **Orchestrations** – private implementation
  - **Choreographies** – public protocols
- Standard-based
- Applicable in any domain – **domain agnostic**
- E.g. the BPM and for scientific computations
- Different industry domains
Service Orchestration: Research Challenges
Research Topics

- **Two orthogonal directions**
- **SBAs**
  - Support for development and execution of SBAs
  - Models, Infrastructures, Adaptation/Flexibility and Monitoring Mechanisms
  - Standardization

- **Apply the knowledge in SOA and SOC**
  - Take into account the overall infrastructure – SW, HW and networks
  - Address challenges of multiple communities
  - Different industries
  - Support multiple business models
SCs and the S-Cube SBA Model
Life Cycle of SBAs and of Service Compositions

1. Identify type of service composition adaptation needed
2. Choose a Service composition adaptation mechanism
3. Identify adaptation need
4. Identify adaptation strategy
5. Enact adaptation
6. Execute a Service composition adaptation mechanism
7. Service composition execution and monitoring
8. Early Requirement Engineering
9. Requirement Engineering & Design
10. Construction
11. Design of service compositions and their interfaces + coordination protocols
12. Operation & management
13. Deployment & Provisioning
S-Cube: SBA Model

- **SBAs: Challenges**
  - Adaptation across layers, Self-* features and QoS-aware SBAs
    - Including adaptation of services, compositions, resources, networks
  - Quality assurance
  - Monitoring
  - Engineering of SBAs

- **On the level of Service Compositions:**
  - **Modelling and execution** of SCs under consideration of BPM requirements (KPIs, rules, governance specifications, etc.)
  - Define a unified, commonly agreed model for **interface** description
  - Capable of accommodating **applications, things, resources, networks**
  - **Adaptation** triggered and mechanism, management of adaptation
  - **Monitoring** of SCs, QoS characteristics and **analys**
Service Compositions and the Future Internet

Outsource

- Reasons
- Models
- Infrastructure

Enterprise ABC

Enterprise ABC

Enterprise XYZ

© Dimka Karastoyanova, 2015
Challenges to Address in the Future

- **Virtualization** of resources, applications and networks
  - A need for **unified service interface description**
  - Focus on **interoperability**

- Tackle **Vertical vs. horizontal orchestration**

- **Provisioning** of services and resources
  - Enabled by service orchestration
  - Provisioning of service orchestrations

- Support for different **business and delivery models**

- **Trust** and **Governance** will have great impact

- Service orchestrations for **networks** to improve coordination and manageability

- Influence Service composition models and infrastructures
  - Relevant to composability of middleware

© Dimka Karastoyanova
Challenges to Address in the Future (2)

- **Cloud computing** and its influence on service orchestrations and enabling infrastructures
  - Service orchestrations for the Cloud
  - Service-oriented applications for Clouds – improved architecture

- **Architecture** of infrastructure for the Future Internet Orchestrations
  - Accommodating services, resources, **networks, contents**

- **Programmability** of Service Orchestrations
  - Simplified programming model enabled by orchestrations
Thanks!
Service Compositions Challenges – Modelling Phase

Choreographies, orchestrations and fragments

Process model

- BPM
- Mapping/ Generation (BPMN → BPEL, BPEL4Chor)
- Mapping from KPIs to PPMs
- Service composition models that contain QoS characteristics
- Service composition languages
- Modelling for adaptation
  - Including fragmentation and coordination protocols
- Modelling for monitoring
  - Event model

PPM, SLAs

Annotate

Process fragment

References to Services

Discovered Services

Mapping from PPMs to QoS

Service Infrastructure

© Dimka Karastoyanova
Service Compositions Challenges – Execution Phase

BPM
- Mechanisms for adaptation
  - Include fragmentation and coordination
- Mechanisms for monitoring

Orchestrations

Processes instances

Includes:
- Fragmentation
- Adaptation
- Coordination

Measurement / Derivation

PPM (combines QoS events and Process-related events)

Service Infrastructure
Challenges in Service Compositions

- Basic assumptions of SOA must be followed
  - Expose things as services via a stable interface
  - Challenge:
    - Define a unified, commonly agreed model for interface description
    - Capable of accommodating applications, things, resources, networks

- Adaptation of compositions
  - Existing service-oriented approaches can be reused
  - Challenge:
    - Adaptation of compositions of networks, resources, things
    - Self-\(^*\), QoS-awareness

- Monitoring and analysis of run time data of compositions
  - Challenge:
    - Mechanisms need to accommodate data from different resources
    - Generic (BAM-like) approaches probably needed
Interoperability has been a challenge.

- BPEL – exchange format – export to and import of BPEL processes into engines → portability
- WSs – interoperability of engines – engines exposed as WSs
- WSs – interactions among processes

**BPEL as exchange format → portability**

Export to BPEL

Import BPEL

WS interaction → engine interoperability

WS interaction → interoperability of processes

BPEL Engine

Workflow Engine