

Service Engineering

Übung 3b – Service-Komposition mit Hilfe von BPEL & Co

Ziele der Übung

- Komposition von Serviceangeboten mit BPEL (Sprachumfang)
- Umgang mit Entwicklungs- und Laufzeitumgebungen (Container)
- Entwicklung eigener eigenen Komposition (BPEL oder BPMN)
- Umgang mit REST-Services und Microservices

Aufgabenstellungen

Aufgabenstellung 1

- Analysieren Sie die folgenden BPEL-Sachverhalte
 - Zielstellungen, Sprachparadigma und Einsatzbereiche
 - Angebotene Sprachelemente (textlich vs. visuell)
 - Vor- und Nachteile der Sprache (Geschäftsprozess-Assembler)
 - Aktuelle Werkzeugunterstützung
- Beschreiben Sie kurz die Eigenschaften der folgenden Elemente:
 - <partnerLinks>
 - <variables>
 - <scope>
 - <sequence> bzw. <flow>
 - Basisaktivitäten und strukturierte Aktivitäten

Aufgabenstellung 2

- Auswahl eines Produkts zur BPEL (ggf. BPMN) Komposition
 - Berücksichtigung vorhandener Kenntnisse
 - Schnelle Möglichkeit zur produktiven Bearbeitung
 - Installation und Test der Funktionsfähigkeit (Beispiel verwenden)

- Implementierung einer eigenen Orchestrierung
 - Erarbeitung eines möglichen Szenarios
 - Einbindung von mind. 2 Serviceangeboten
 - exakte Dokumentation der notwendigen Schritte

Aufgabenstellung 3

- A: Umgang mit RESTful Serviceangeboten unter BPEL (optional)
 - Berücksichtigung in der aktuellen BPEL-Spezifikation?
 - Analyse möglicher Ansätze zur Orchestrierung von REST Services?
 - Bewertung der Sinnfälligkeit entsprechender Ansätze?
- B: (SOA-) Orchestrierung vs. Microservices (optional)
 - Verständnis SOA-Service und Microservice?
 - Vergleich technischer, architektonischer und organisatorischer Aspekte?
 - Diskussion der Orchestrierung und Choreografie bei Microservices?

Bem.: Bei Aufgabenstellung 3 ist mindestens eine (A oder B) zu wählen!

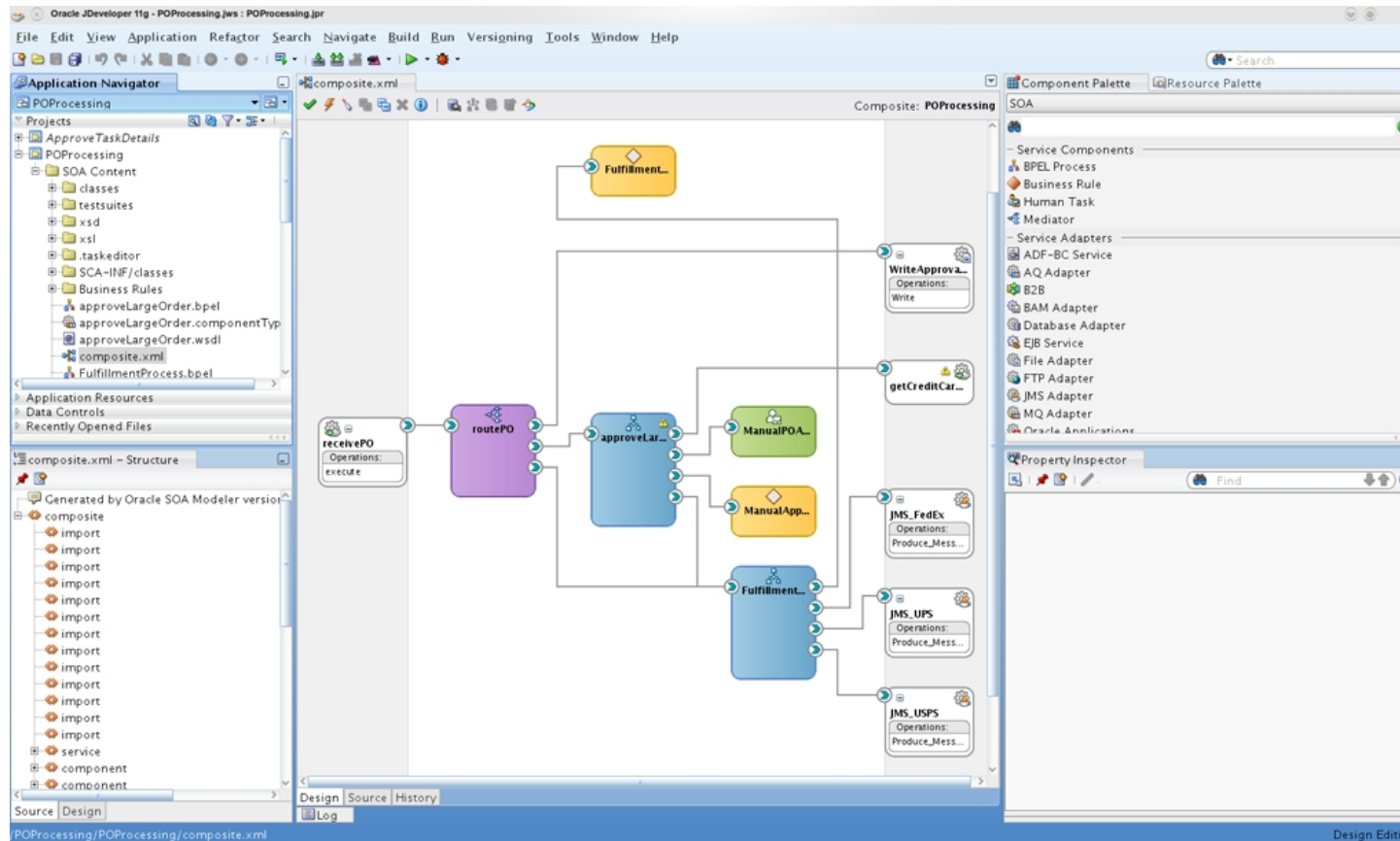
Weiterführende Hinweise

Aufbau einer BPEL-Datei

```
<?xml version="1.0"?>
<process name="aendereAdresse" ...>
  <variables>
    <variable messageType="..." name="...">
  </variables>
  <flow>
    <receive .../> <!-- bei diesem Input (OP-Name, Params) -->
    <invoke .../> <!-- anderen Service aufrufen -->
    <assign .../> <!-- Daten mappen -->
    <reply .../> <!-- Antwort(-Daten) liefern -->
  </flow>
</process>
```

Quelle: Josuttis, N.: SOA in der Praxis, S. 113, dpunkt.verlag GmbH, Heidelberg 2009

Beispiel: Oracle BPEL Process Manager



Quelle: <http://www.oracle.com/technetwork/middleware/bpel/overview/index.html>, Abruf: März 2017

Beispiel: jBPM BPMN-Ausführung in Eclipse

```
ProcessTest.java
TaskService taskService = engine.getTaskService();

// start a new process instance
Map<String, Object> params = new HashMap<String, Object>();
params.put("employee", "krisyv");
params.put("reason", "Yearly performance evaluation");
ProcessInstance processInstance =
    ksession.startProcess("com.sample.evaluation", params);
System.out.println("Process started ...");

// complete Self Evaluation
List<TaskSummary> tasks = taskService.getTasksAssignedAsPotentialOwner("krisyv", "en-UK");
assertEquals(1, tasks.size());
```

Console Output:

- Process started: Evaluation[com.sample.evaluation]
- Process node triggered: in process Evaluation[com.sample.evaluation]
- Process node triggered: Self Evaluation in process Evaluation[com.sample.evaluation]
- Process node triggered: in process Evaluation[com.sample.evaluation]
- Process node triggered: PM Evaluation in process Evaluation[com.sample.evaluation]
- Process node triggered: HR Evaluation in process Evaluation[com.sample.evaluation]
- Process node triggered: in process Evaluation[com.sample.evaluation]
- Process node triggered: in process Evaluation[com.sample.evaluation]

Quelle: <http://www.jbpm.org/>, Abruf: März 2017

Herausforderung – Verständnis Microservice



Microservices Resource Guide

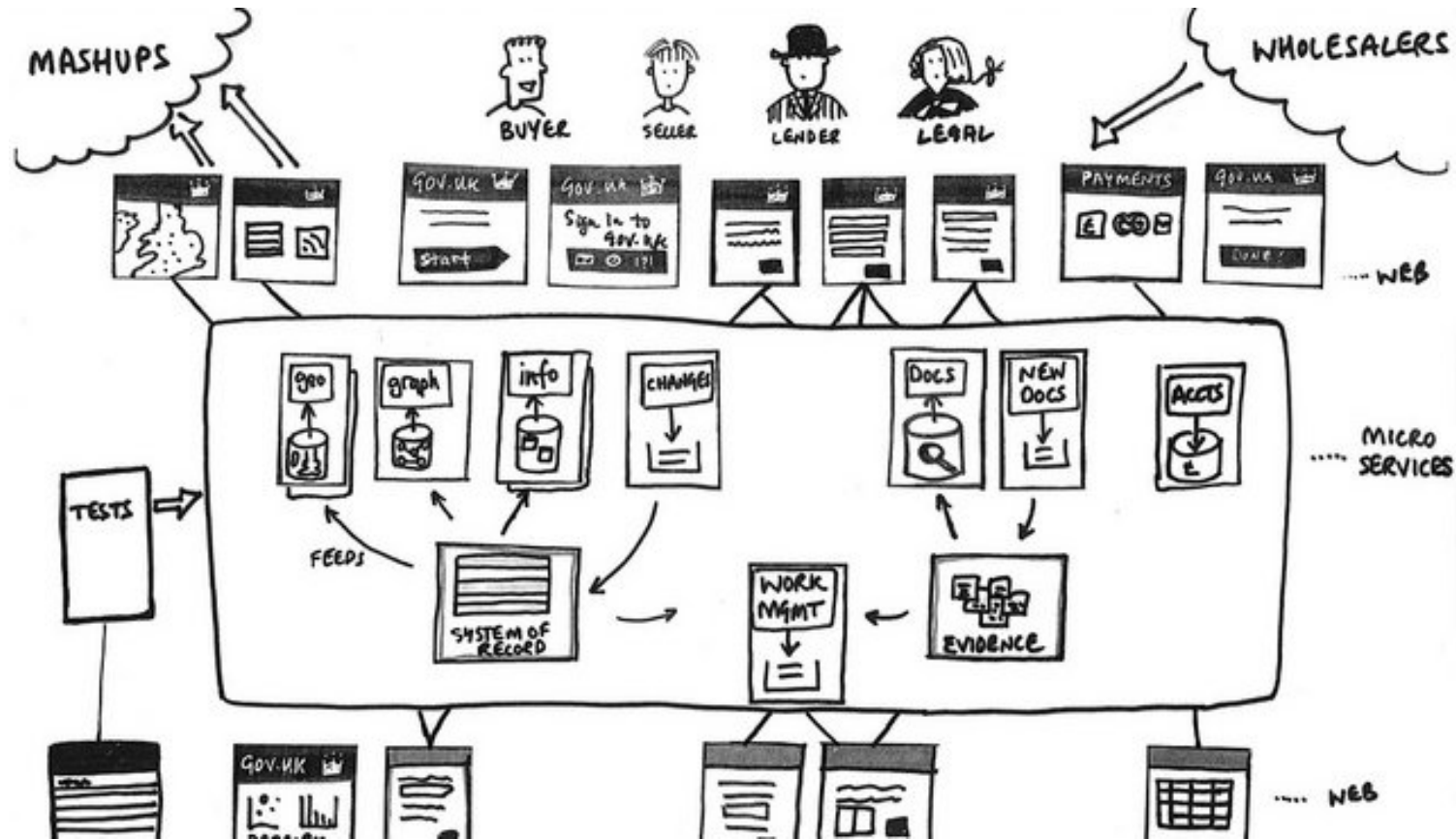
"Microservices" became the hot term in 2014, attracting lots of attention as a new way to think about structuring applications. I'd come across this style several years earlier, talking with my contacts both in ThoughtWorks and beyond. It's a style that many good people find is an effective way to work with a significant class of systems. But to gain any benefit from microservice thinking, you have to understand what it is, how to do it, and why you should usually do something else.



This is a guide to useful resources to find out more about microservices. It's a personal choice of articles, videos, books, and podcasts that can teach you more about the microservices architectural style. Primarily the selection helps to guide you through material on martinowler.com, but I also cover additional material that I think is valuable. It does not attempt to be a comprehensive list (I've selected just a couple of introductory talks from the many out there) but this is my suggestion for places where your exploring should start.

Quelle: <https://martinfowler.com/microservices/>, Abruf: März 2017

Microservices Orchestration Design Patterns



Quelle: Gupta, M.: Microservices Orchestration Design Patterns, April 2015, <https://www.linkedin.com/pulse/microservices-orchestration-design-patterns-munish-kumar-gupta/>, Abruf: März 2017

Empfohlene Literaturquellen

- Josuttis, N.: SOA in der Praxis – Systemdesign für verteilte Geschäftsprozesse, dpunkt.verlag GmbH, Heidelberg 2009
- Tilkov, S.; Eigenbrodt, M.; Schreier, S.; Wolf, O.: REST und HTTP, dpunkt.verlag GmbH, Heidelberg 2015
- Wolf, E.: Microservices – Grundlagen flexibler Softwarearchitekturen, dpunkt.verlag GmbH, Heidelberg 2016
- Newman, S.: Building Microservices: Designing Fine-Grained Systems 1st Edition, O'Reilly, Sebastopol, USA/CA 2015

Organisation der Übung

Aufbau des Protokolls

Verwenden Sie bitte des folgende Muster für das Protokoll:

- Allgemeine Angaben
 - Versuch, Beteiligte Studenten, Datum
 - Rahmenbedingungen (Hardware, Software, ...)
 - Methodisches Vorgehen zu Bearbeitung
- Aufgaben des Laborversuchs
 - Entsprechend der Struktur der Aufgabenstellung
 - Textliche Ausführungen zu den Lösungen
 - Verwendung von Grafiken und Screenshots
- Zusammenfassung (Bewertung der erreichten Ergebnisse)
- Genutzte Quellen (z.B. Literatur, Internet, ...)

Organisation der Übung

Bitte berücksichtigen Sie die folgende Vorgehensweise:

- Die Übung wird in Gruppen durchgeführt!
- Vorgeschlagene Methode zur Aufgabenlösung
 - Abstimmung einer geeigneten Vorgehensweise (15 min)
 - Ausführen der Aufgabenstellungen (120 min)
 - Erstellung eines Protokolls & Präsentation (45 min)
 - Präsentation der Ergebnisse (je Gruppe – 20 min inkl. Fragen)
- Gesamtzeit für die Durchführung: ca. 180 min
- Bereitstellung eines entsprechenden Protokolls je Gruppe