

# **Logistik Service Engineering und Management:** Modellgetriebene ad-hoc Integration von Logistikdienstleistern - Integrationsansatz und Prototyp

Robert Kunkel  
Mittwoch, 9. Mai 2012  
Entwicklertage 2012

# Agenda

- Motivation
- LSEM
- Modellgetriebene ad-hoc Integration
- Integrationsvarianten
- Modellgetriebene Integration
- Integrationsmodell
- Prototyp

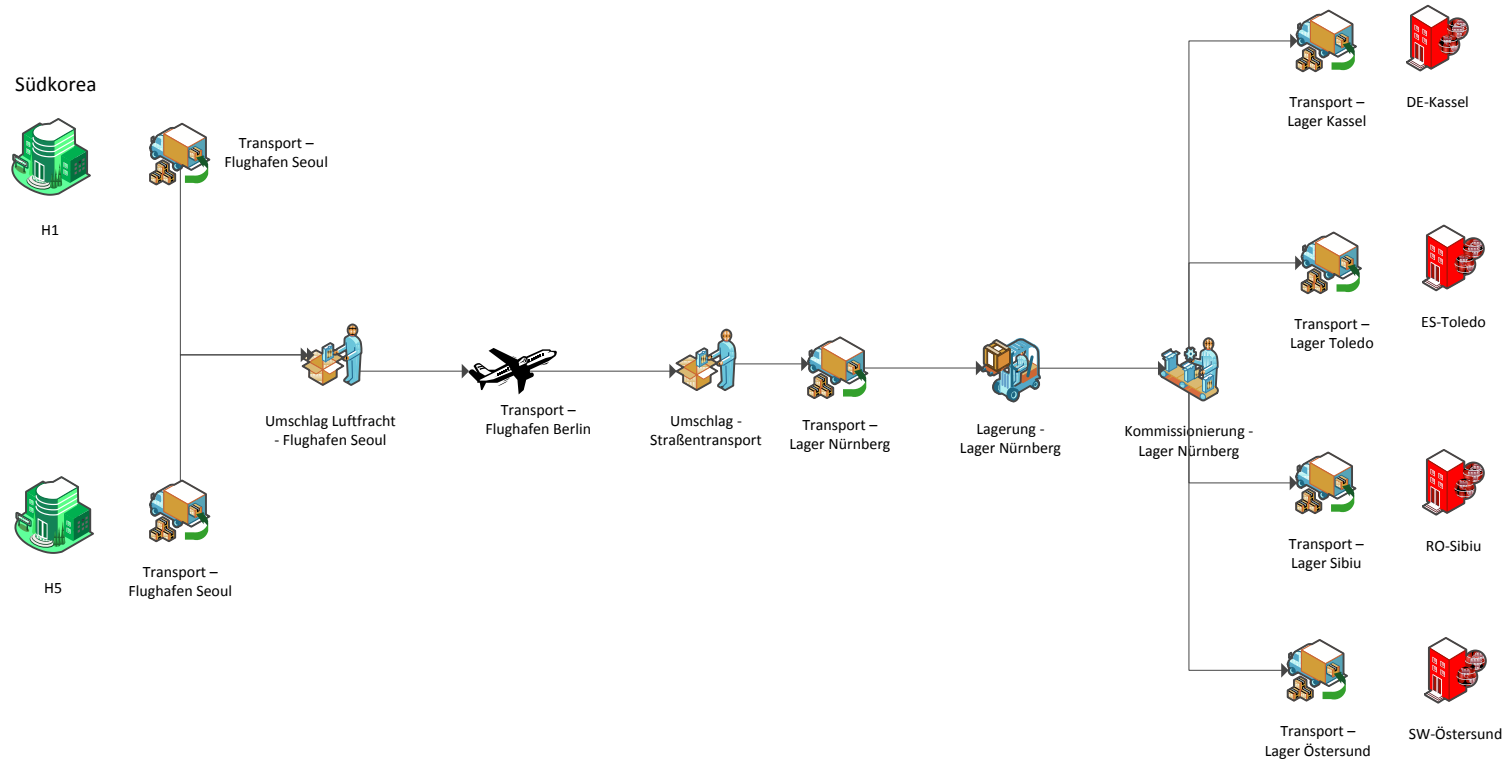
# Motivation

- Logistik ist mit **210 Mrd. Euro Umsatz** und einer **Beschäftigung von ca. 2,6 Mio.** Menschen einer der wichtigsten Dienstleistungsbereiche in Deutschland
- Deutscher Logistikmarkt wird im europäischen Vergleich **überdurchschnittlich stark wachen** [Korschinsky 2010]
- **Region Mitteldeutschland** als Logistikkreuz

# Motivation

- **Logistik-Outsourcing:** Logistikabteilungen, Outsourcing einzelner Dienstleistungen, Joint-Venture, Kontraktlogistik
- Ausgelagert werden die klassischen logistischen Dienstleistungen **Transport, Umschlag, Lagerung** aber auch **zahlreiche Zusatzbereiche** wie Verpackung, Veredelung, Verzollung oder Reifung
- **Mehrwertlogistikdienstleister** bzw. Fourth Party Logistics (4PL)
  - Managementkonzept: Integration, Planung, Steuerung und Kontrolle von Güter- und Informationsflüssen über Logistikdienstleister hinweg (vgl. [Bauknight, 1999])

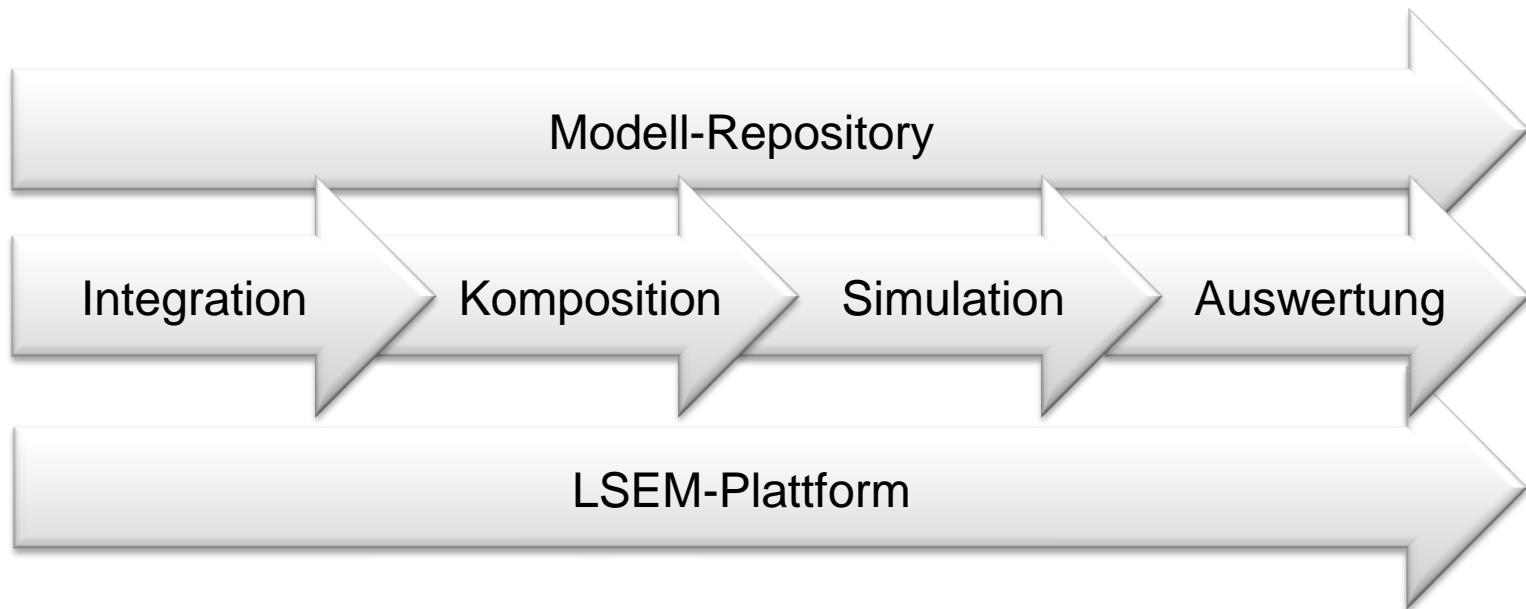
# Beispielszenario MLDL



- Die effiziente Bereitstellung von Informationen kann den Warenfluss in einer Lieferkette deutlich beschleunigen (vgl. [14])

- Forschungsprojekt **LSEM**: Logistik Service Engineering und Management
  - Das Gesamtziel von LSEM besteht darin, **Mehrwert-LDL** sowie Logistik-anbietenden und -auslagernden Unternehmen eine **konkrete Unterstützung** (Modelle, Methoden, Werkzeuge) für das Engineering und Management von **arbeitsteiligen Mehrwertlogistikdienstleistungen** zu geben
  - Gemeinsames Gestaltungsparadigma von **Business Services** (formalisierte fachliche LD) und **Electronic Services** (technische Dienstleistungsangebote)
    - Integrierte gemeinsame Sichtweise
    - Zusammensetzung zu komplexen Mehrwert-LD
  - Zusammenarbeit mit Netzwerk Logistik Leipzig/Halle e.V mit > 100 Unternehmen

# LSEM – Bereiche



# Architektur LSEM-Plattform

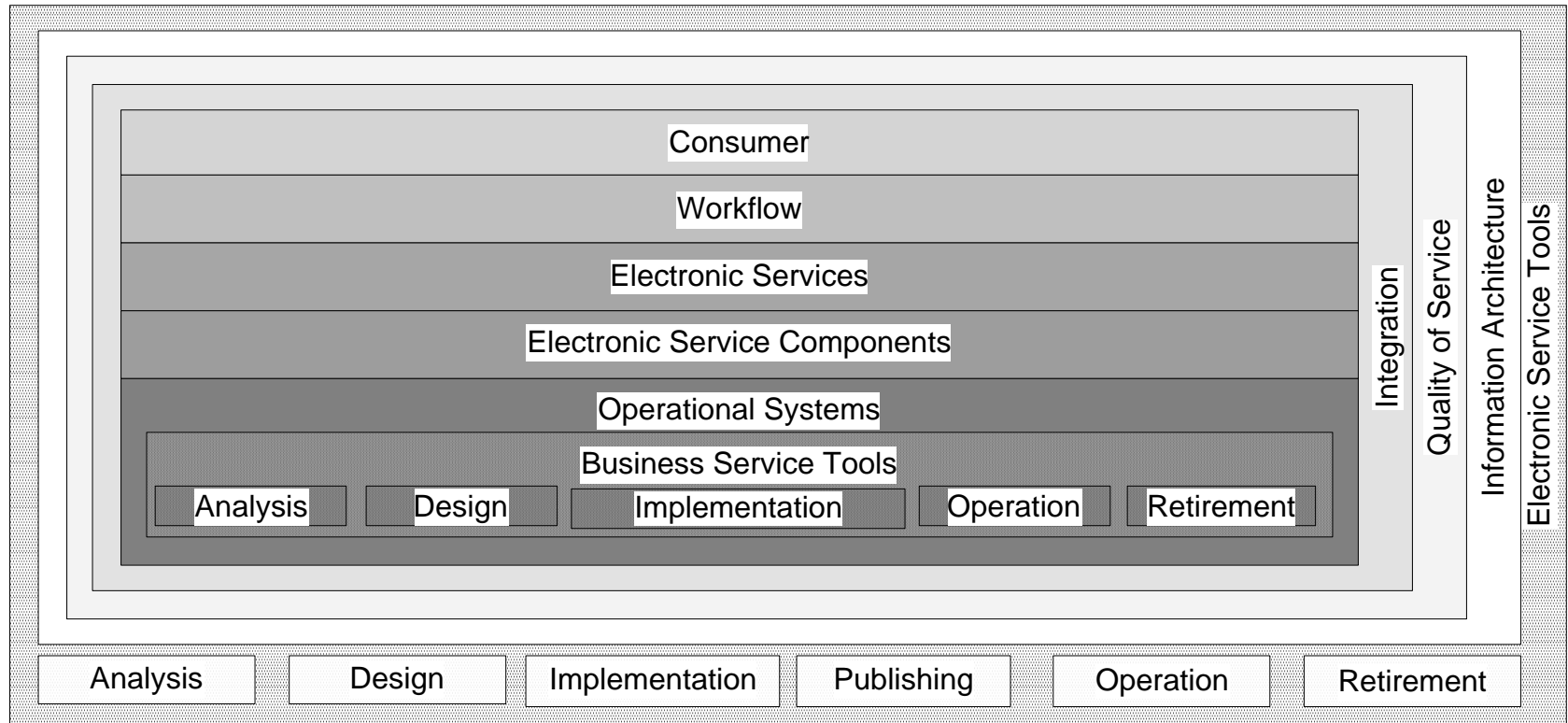
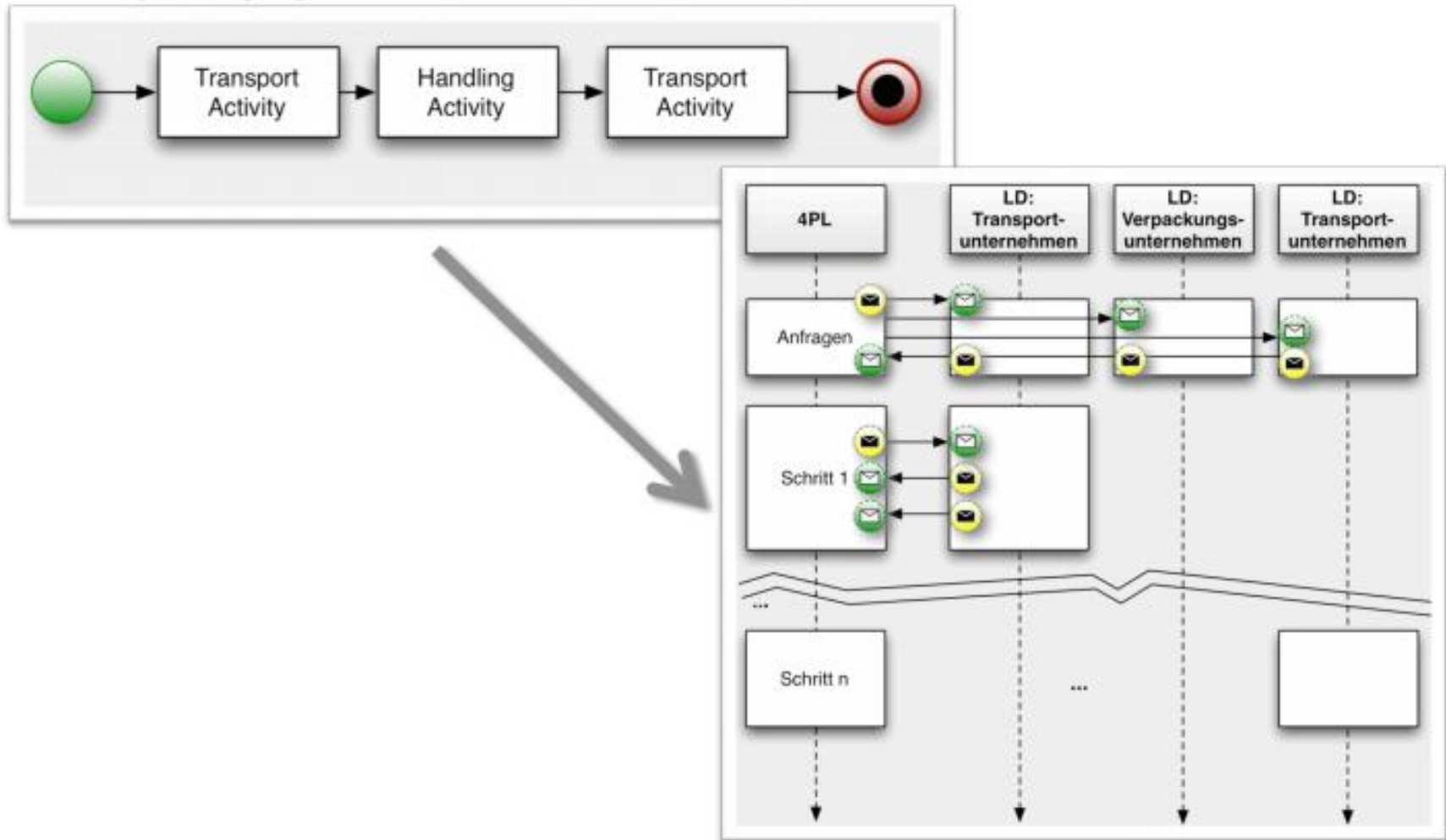


Abbildung 1: Die Architektur der LSEM-Plattform (aufbauend auf der S3-Referenzarchitektur vgl. [1] und [11])

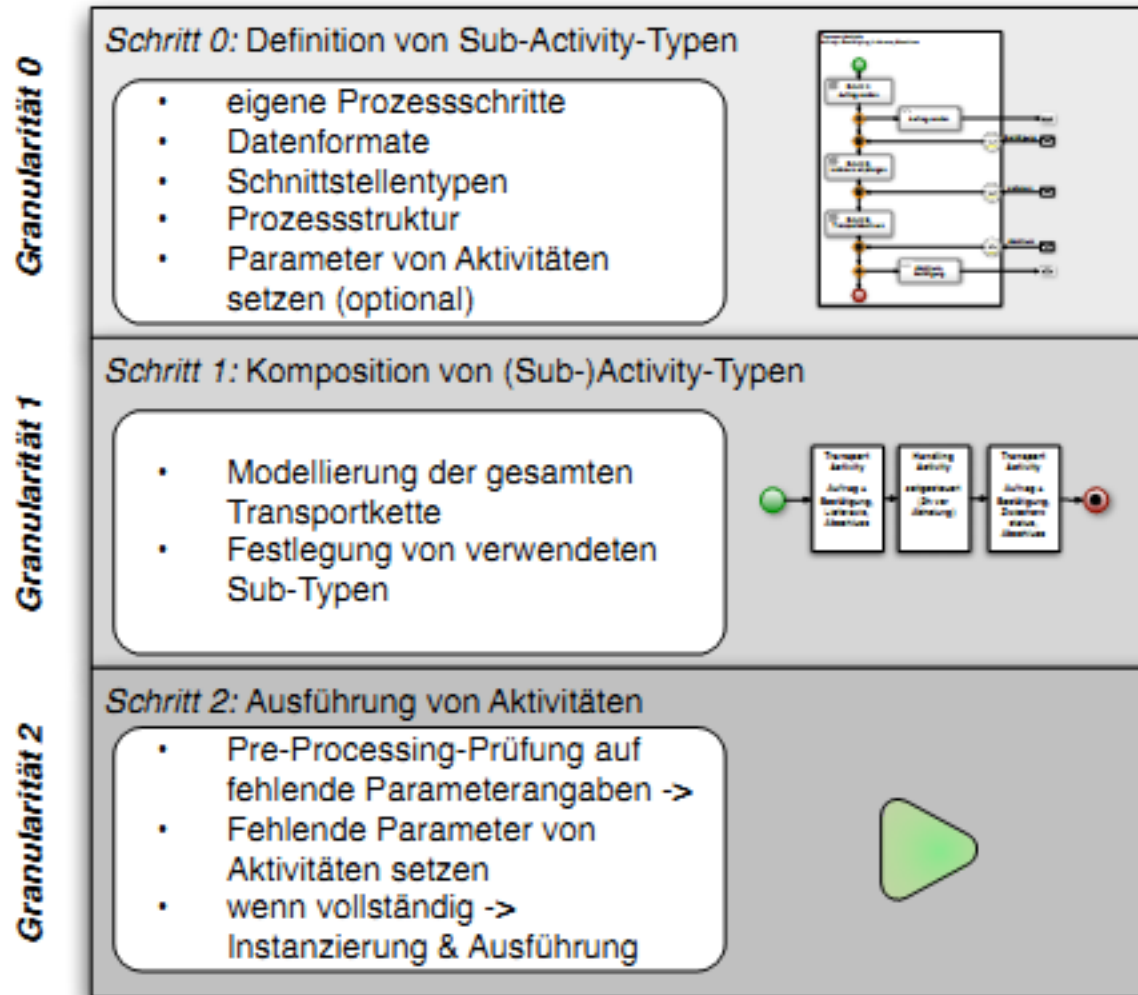


# **Modellgetriebene ad-hoc Integration von Logistikdienstleistern - Integrationsansatz und Prototyp**

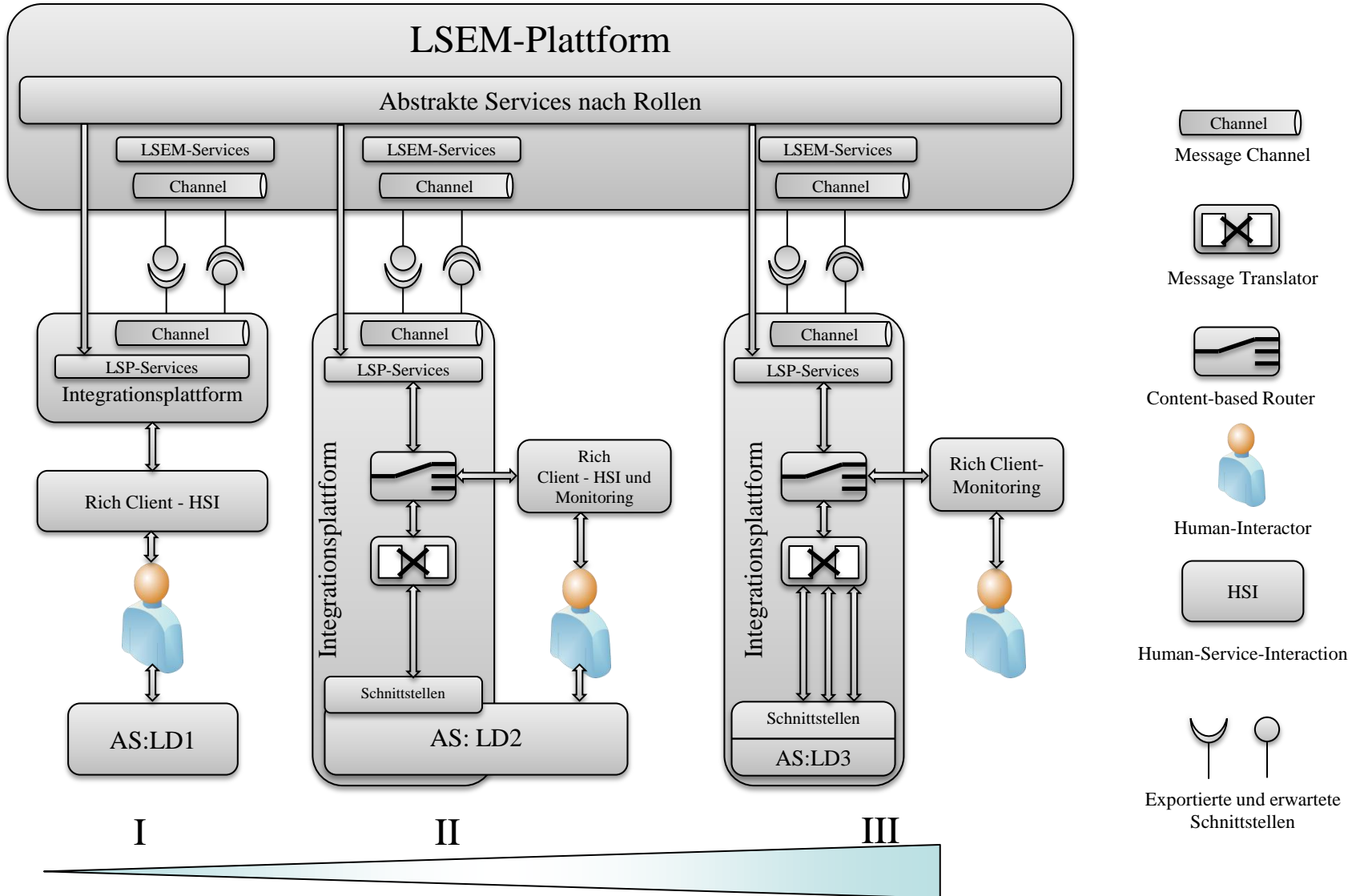
# Ausgangsbasis und Ziel



# Granularitäten



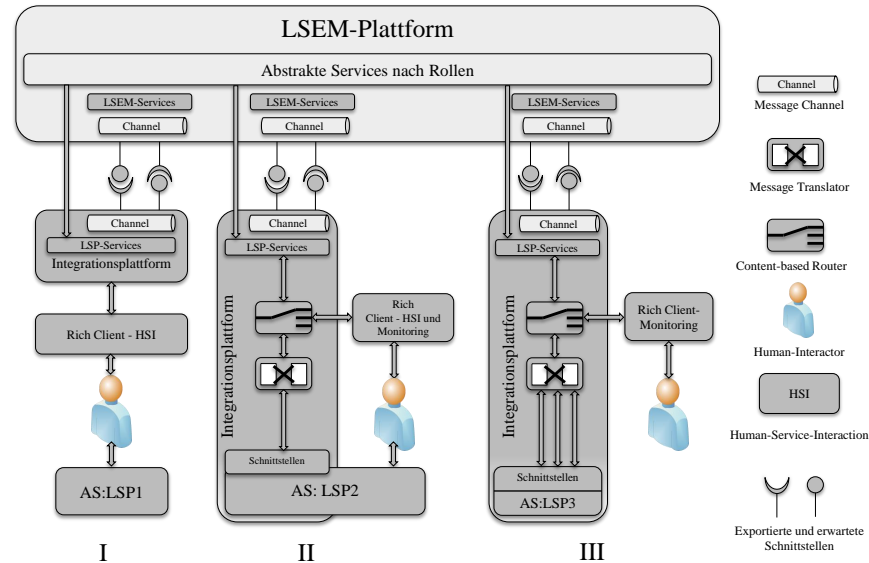
# Integrationsvarianten



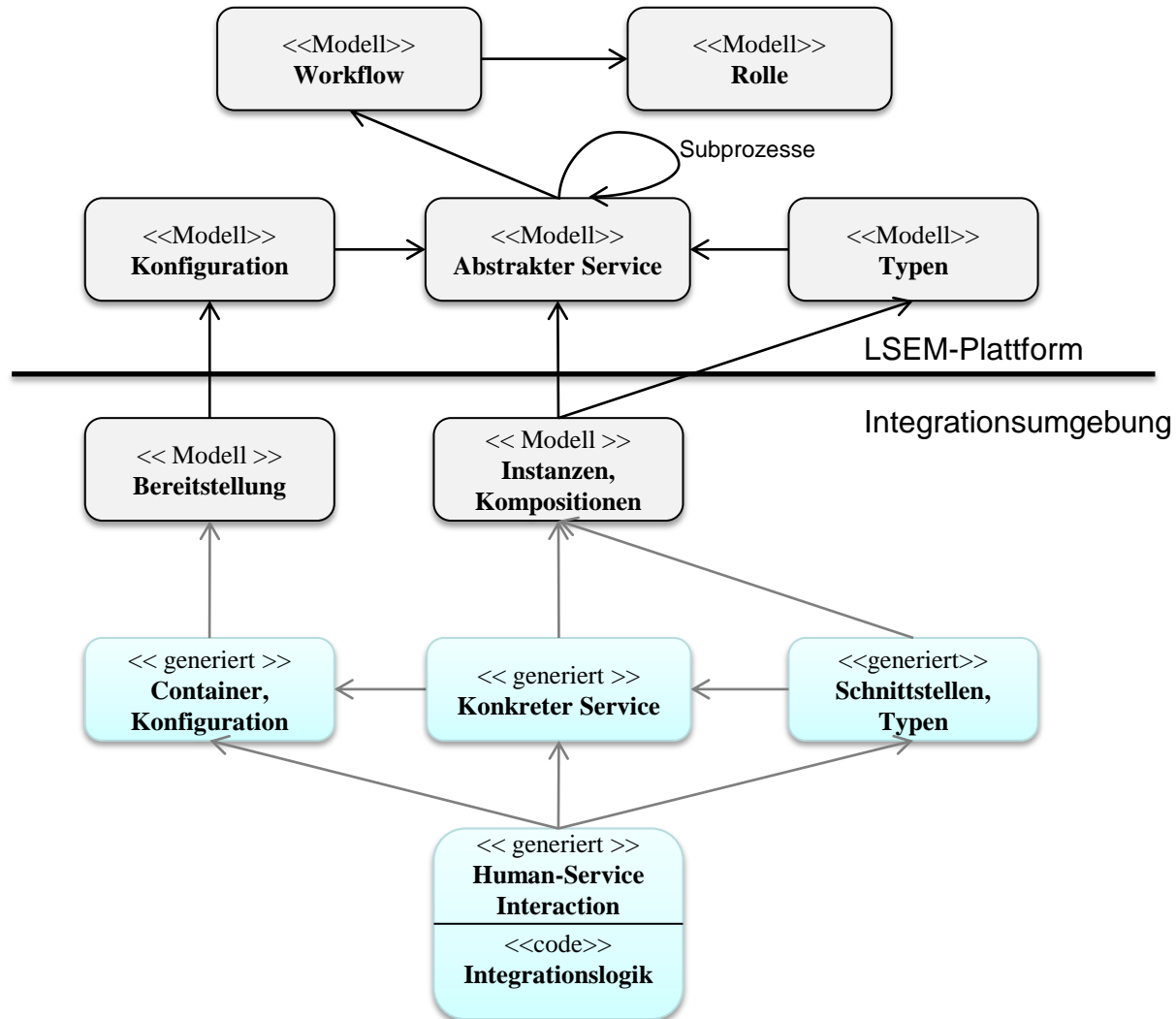
Integrations- und Automatisierungsgrad & Integrationsaufwand

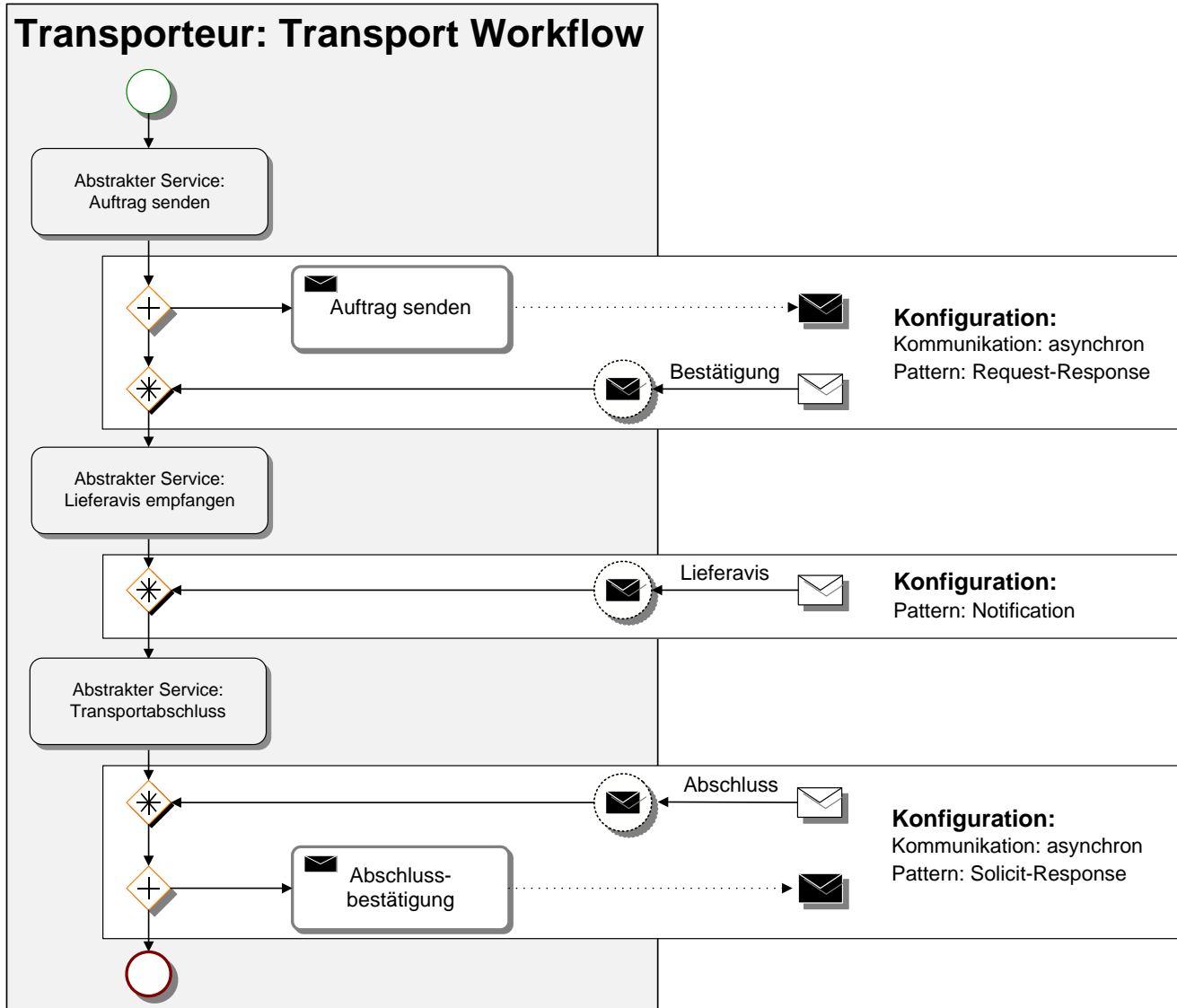
Vgl. [Kunkel, 2012]

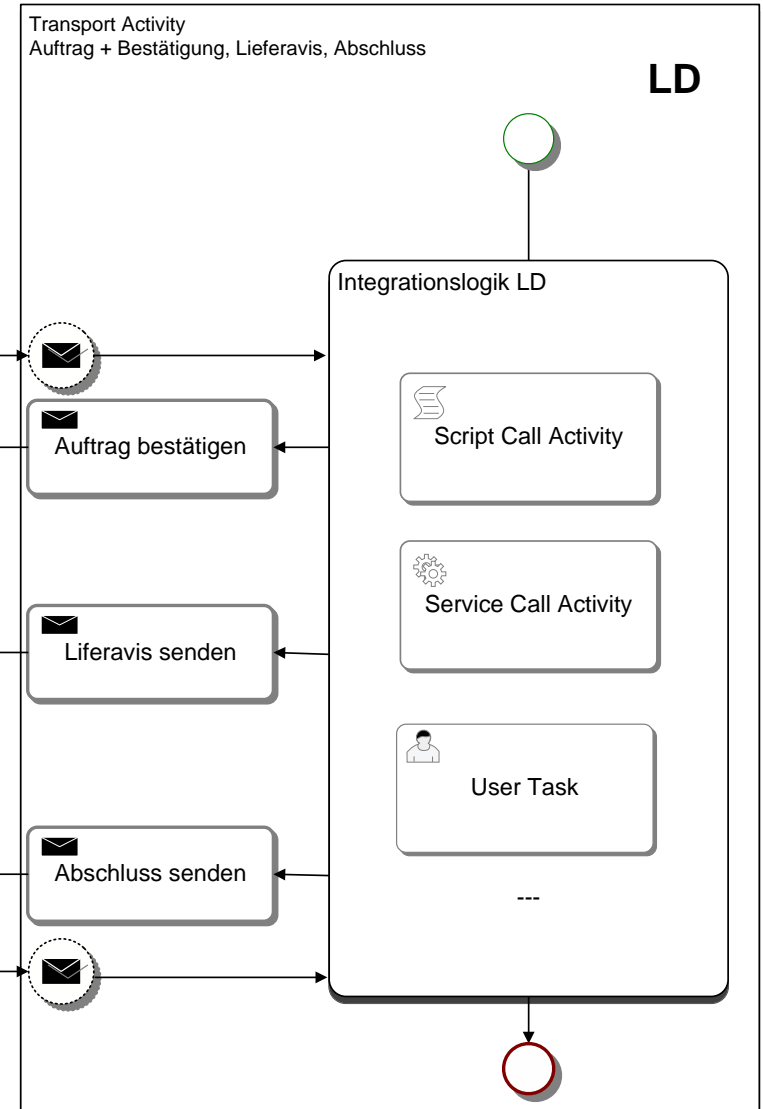
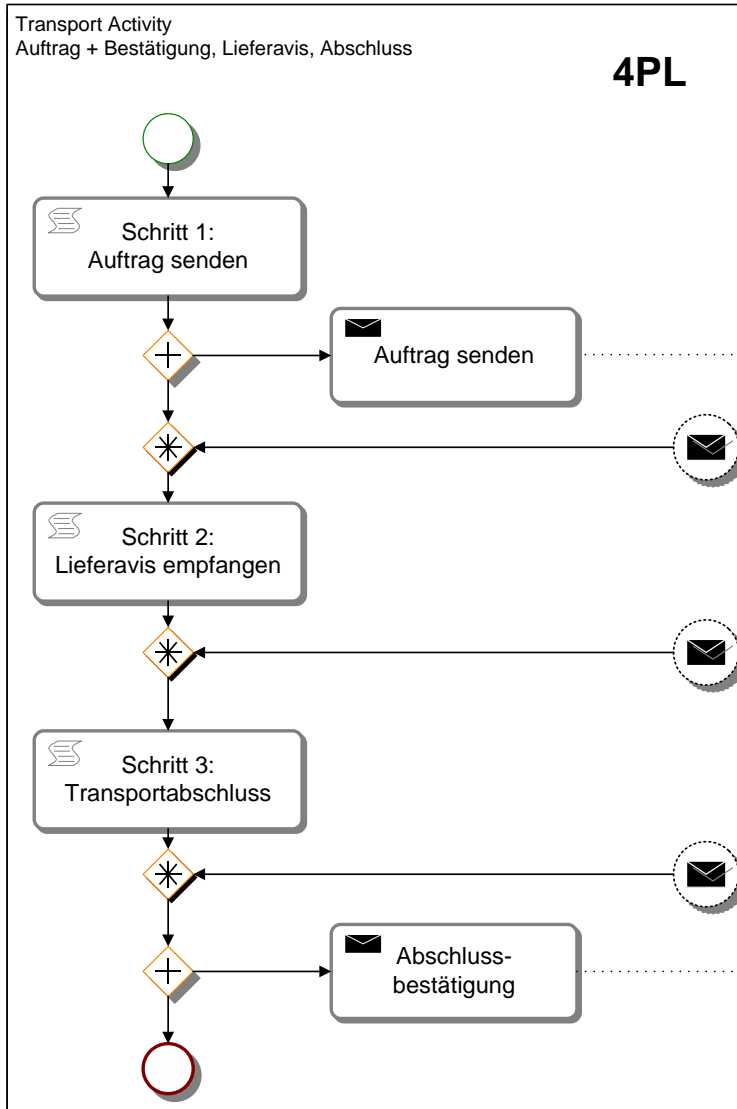
# Modellgetriebene ad-hoc Integration



- Um die effiziente (ad-hoc) Bereitstellung einer Integrationsumgebung zu ermöglichen eignen sich vor allem modellgetriebene Verfahren, da diese besondere Stärken aufweisen, wenn mehrere verwandte Produkte entwickelt werden und Teile sich wiederverwenden lassen vgl. [Stahl, 2007] und [Thränert, 2008]
- Komplexität lässt sich auf das Modell selbst und auf die Abbildung des Modells auf die Zielsprache aufteilen

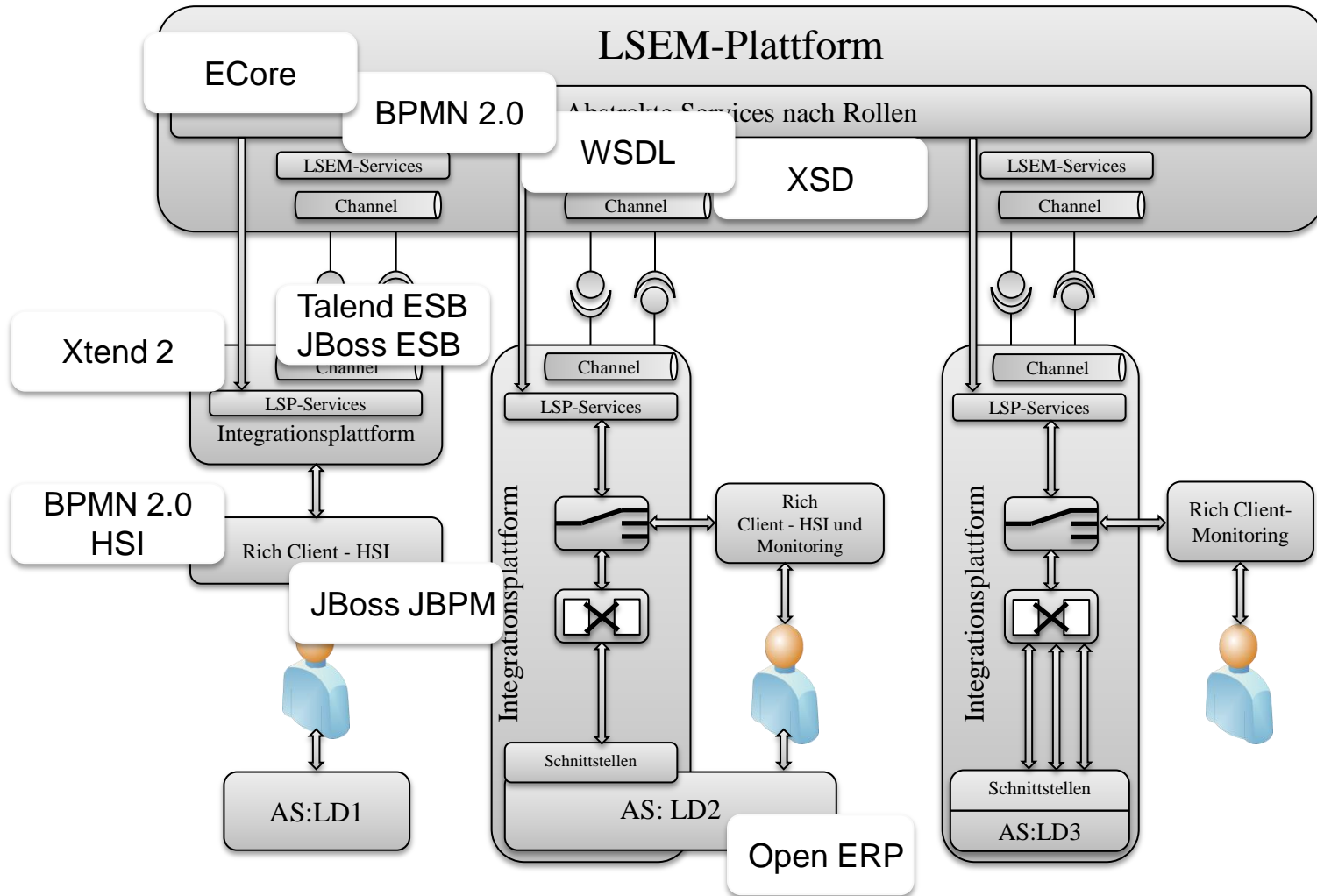








# Prototyp



The screenshot shows the Drools IDE interface. On the left is a 'Knowledge Bases' tree with a 'Browse' button. The tree structure is as follows:

- Knowledge Bases
  - Portground
  - Repository
    - HandlingServiceCategory
    - Schemas
      - GS1
        - Transport
          - CapacityBookingAndResponse
          - CapacityPlan
          - CapacityRequirements
          - InstructionAndResponse
          - PickUpDropOffRequestAndConfirmation
          - StatusRequestAndNotification
        - sbdh
        - shared
      - ServiceDefinitions
      - SpecialServiceCategory
      - Test
    - TransportationServiceCategory
    - ValueAddedServiceCategory
    - WarehouseServiceCategory
  - Simulation
  - UniLeipzig
- com
- de
  - defaultPackage
  - guvnor/BPM
- mdie
- Global Area

The main workspace shows a table of assets:

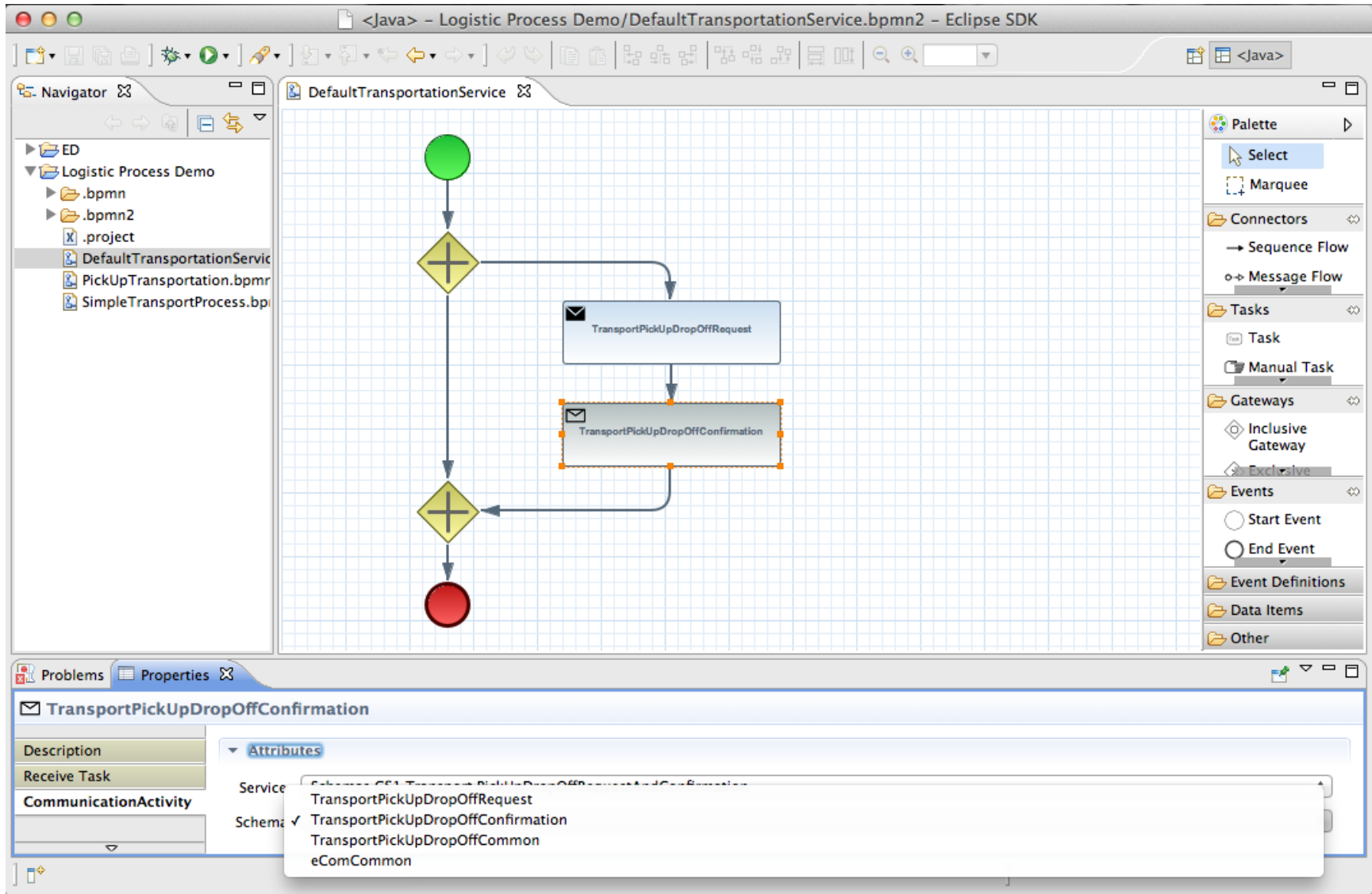
Format	Name	Status	Last modified	Open
<input type="checkbox"/>	TransportPickUpDropOffCommon	Draft	2012 Mar 7 17:54:51	<input type="button" value="Open"/>
<input type="checkbox"/>	TransportPickUpDropOffConfirmation	Draft	2012 Mar 7 17:55:32	<input type="button" value="Open"/>
<input type="checkbox"/>	TransportPickUpDropOffRequest	Draft	2012 Mar 7 17:55:50	<input type="button" value="Open"/>
<input type="checkbox"/>	eComCommon	Draft	2012 Mar 7 17:53:50	<input type="button" value="Open"/>

At the bottom of the workspace, there are buttons for 'QA', 'Package snapshots', and 'Administration'. A 'Close all items' button is located at the bottom right of the workspace area.

The screenshot displays the VDF - Eclipse Platform interface. On the left, the 'VDF Navigator' shows a project structure for 'VDF-Example' with folders for '00. Branch', '01. Process', '02. Dataformat', '03. Transformation', and '04. Communication'. Under '04. Communication', there is a 'DataBase' folder containing an 'index' folder and several files related to 'neostore'. The main area shows a 'Database graph' with a root node 'label: Vertical Domain Facilities' connected to four child nodes: 'label: Domains', 'label: Dataformat', 'label: Process', and 'label: Communication'. The relationships are labeled 'DOMAIN\_ROOT', 'DATAFORMAT\_ROOT', 'PROCESS\_ROOT', and 'COMMUNICATION\_ROOT'. A 'SUBDOMAIN' relationship connects 'label: Domains' to 'label: Transportation'. The 'label: Transportation' node has properties: 'type: Domain' and 'uri: http://devels.wifa.uni-leipzig.de:8080/Transportation/Transportation'. At the bottom, the 'Properties' and 'Relationship types' panels are visible.

Property	Value
derived	false
editable	true
last modified	28. Februar 2012 18:36:33
linked	false
location	C:\Users\brakasa\workspace\VDF-Example\VDF-Example\Data...
name	DataBase

Relationship type	In	Out
COMMUNICATION_ROOT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DATAFORMAT_ROOT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DOMAIN_ROOT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PROCESS_ROOT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SUBDOMAIN	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



The screenshot displays the Eclipse IDE interface for editing a BPMN diagram. The main workspace shows a process flow starting with a green circle, followed by a task named 'DefaultTransportationService' (highlighted with a dashed orange border), then a task named 'Annahme Fracht', and ending with a red circle. The left sidebar shows a project tree with 'Logistic Process Demo' containing '.bpmn', '.bpmn2', and '.project' folders, and several BPMN files. The right sidebar shows a palette with various BPMN elements like Connectors, Tasks, Gateways, and Events. The bottom pane shows the properties for the selected 'DefaultTransportationService' task, including 'Logistic Type' and 'SubProcess'.

**DefaultTransportationService**

Description	Attributes
Task	
Logistic Activity	
Logistic Type	TransportationService
SubProcess	DefaultTransportationService

The screenshot shows the Eclipse IDE interface for a BPMN process. The main editor displays a process diagram with a green start event, a task named 'DefaultTransportationService', another task named 'Annahme Fracht', and a red end event. A context menu is open over the diagram, showing options: '(no launch history)', 'Run As' (with a submenu containing '1 Run as Logistic Process'), 'Run Configurations...', and 'Organize Favorites...'. The left sidebar shows a project tree with files like 'SimpleTransportProcess.bpmn'. The right sidebar contains a palette with various BPMN elements like 'Start Event', 'End Event', and 'Task'. The bottom panel shows the 'Default Process' description: 'A Process Diagram describes a sequence or flow of Activities in an organization with the objective of carrying out work. A Process is depicted as a graph of Flow Elements, which are a set of Activities, Events, Gateways, and Sequence Flows that define finite execution semantics. Processes can be defined at any level from enterprise-wide Processes to Processes performed by a single person. Low-level Processes can be grouped together to achieve a common business goal.'

# Prototyp

The screenshot displays a software development environment with two main windows. The left window shows a metadata tree for a file named 'TransportOrderData.mdie'. The tree structure is as follows:

- TransportOrderData.mdie
  - NameAndAddressSimpleType.xsd
  - FreightItems.xsd
  - pickupLocation
    - sequence
      - longitude : xs:float
      - latitude : xs:float
      - pickup : xs:dateTime
      - address : Isb:NameAndAddressSimpleType
  - deliveryLocation
    - sequence
      - longitude : xs:float
      - latitude : xs:float
      - delivery : xs:dateTime
      - address : Isb:NameAn
  - transportOrderItem
    - sequence
      - pickupLocation : tns:p
      - deliveryLocation : tns:
      - freightItem : Isbf:freig
  - TransportOrder
    - sequence
      - transportOrderItem : t

The right window shows a BPMN process diagram for a process named 'Transport'. The process starts with a start event, followed by a task 'Auftrag senden'. This leads to an AND-split gateway, which branches into two paths: one leading to a message event 'sende Auftrag' and another to an AND-join gateway. The 'sende Auftrag' event leads to a task 'Bestätigung', which then leads to a message event 'Lieferavis'. This event leads to an AND-join gateway, which then leads to a task 'Transportabschluss'. This task leads to a message event 'Rechnung', which leads to another AND-join gateway. This gateway leads to a task 'Abschluss senden', which finally leads to an end event. The process information panel on the left of the diagram window shows the following details:

Process Information  
Name: Transport  
Format: bpmn2  
Package: mdie.hts  
Created: 2012-02-29T14:38:55.773+01:00  
Created By: superUser  
Last Modified: 2012-03-01T12:15:59.260+01:00  
Comment:  
Version: 4

The image shows a screenshot of a BPMN editor window titled "DisposeTransportOrderOperation.bpmn". The window is split into two panes. The top pane displays the XML code for the process definition, and the bottom pane displays the corresponding BPMN diagram.

**XML Code (Top Pane):**

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions id="Definition"
targetNamespace="http://www.jboss.org/drools"
typeLanguage="http://www.java.com/javaTypes"
expressionLanguage="http://www.mvel.org/2.0"
xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.omg.org/spec/BPMN/20100524/MODEL BPMN20.xsd"
xmlns:g="http://www.jboss.org/drools/flow/gpd"
xmlns:bpmdi="http://www.omg.org/spec/BPMN/20100524/DI"
xmlns:dc="http://www.omg.org/spec/DD/20100524/DC"
xmlns:di="http://www.omg.org/spec/DD/20100524/DI"
xmlns:tns="http://www.jboss.org/drools">
<itemDefinition id="_TransportOrderItemItem" structureRef="java.lang.String" />
<itemDefinition id="_TransportUnitsItem" structureRef="java.lang.String" />
<itemDefinition id="_DriversItem" structureRef="java.lang.String" />
<itemDefinition id="_contentItem" />

<process processType="Private" isExecutable="true" id="com.sample.bpmn.DisposeTransportOrderOperation" name="Process:

<extensionElements>
<tns:global identifier="result" type="java.util.Map" />

</extensionElements>
<property id="TransportOrderItem" itemSubjectRef="_TransportOrderItemItem" />
<property id="TransportUnits" itemSubjectRef="_TransportUnitsItem" />
<property id="Drivers" itemSubjectRef="_DriversItem" />
  <property id="content" itemSubjectRef="_contentItem"/>

<startEvent id="_1" name="StartProcess" />
<scriptTask id="_6" name="InputVariablen von WS kopieren" scriptFormat="http://www.java.com/java" >
<script>java.util.Map contentParam = new java.util.Hash
contentParam.put ("TransportOrderItem", TransportOrderItem
contentParam.put ("TransportUnits", TransportUnits);
contentParam.put ("Drivers", Drivers);
kcontext.setVariable ("content", contentParam);
</script>
  <scriptTask
    <userTask id="_8" name="HSI" >
      <ioSpecification>
        <dataInput id="8_ContentInput" name="Content" />

```

**BPMN Diagram (Bottom Pane):**

The diagram shows a linear process flow:

- Start Event:** A green circle.
- Script Task:** A yellow rounded rectangle labeled "InputVari...".
- User Task:** A yellow rounded rectangle labeled "HSI".
- End Event:** A red bullseye.

Sequence flows connect these elements in order. A toolbar on the left of the diagram pane includes options like "Select", "Marquee", "Sequence Flow", "Components", "Start Event", "End Event", "Service Tasks", "Log", and "Email".



```
<html>
<body bgcolor="#BFBFBF">
<h2>HumanTask!</h2>
<table border="1" bordercolor="#000000" style="background-color:#CCCCCC" width="400" cellpadding="3" cellspacing="3">
<tr>
<td>TransportOrderIter
</tr>
<tr>
<td>TransportUnits:</td>
<tr>
<tr>
<td>Drivers:</td><td>
<tr>
</table>
<form action="complete"
enctype="multipart/form-data"
<table border="1" bordercolor="#000000" style="background-color:#CCCCCC" width="400" cellpadding="3" cellspacing="3">
<br>
<tr><td>Annehmen:</td>
<br>
</table>
<input value="Complete" type="button" />
</form>
</body>
</html>
```

Tasks  
Personal Tasks  
Group Tasks

john Logout

### Task Form: AnbieterJohn

Dieser Auftrag wurde angefragt!

Von: Firma: HiQ-Electronics  
Adresse: Makolm Spence Street  
9000 Windhoek (Namibia)

Nach: Firma: Airport Africa AG  
Adresse: Airportstreet  
9000 Windhoek (Namibia)

Abholzeitspanne: frühester Termin: 27.08.2010 02:21Uhr  
spätester Termin: 27.08.2010 02:21Uhr

Lieferzeitspanne: frühester Termin: 29.08.2010 02:21Uhr  
spätester Termin: 29.08.2010 12:00Uhr

Waren: 33 Stück: DVD Player X1860

Annehmen Ablehnen

Processes  
Reporting  
Settings

- Live-Demos im LSB-Demo-Labor
- Nach Umbau ab Februar 2013 im LSEM Living Lab
- <http://www.wifa.uni-leipzig.de/islog/forschung/living-lab.html>



# Quellen

- (1) Arsanjani, A, et al. (2007): S3: A Service-Oriented Reference Architecture. IT Professional 9:10-17.
- (2) Bauknight, D, Miller, J (1999): Fourth Party Logistics: The Evolution of Supply Chain Outsourcing. In: CALM Supply Chain & Logistics Journal.
- (3) Benatallah, B, et al (2005): Developing Adapters for Web Services Integration. 17th International Conference Advanced Information Systems Engineering. Porto.
- (4) Bowen, D (1986): Managing customers as human resources in service organizations. In: Human Resource Management 25(3): 371-383, Wilmington.
- (5) Bussler, C (2003): B2B Integration: Concepts and Architecture. Springer, Berlin.
- (6) Dix, A, et al. (2003): Human-Computer Interaction. Prentice Hall, Harlow.
- (7) Erl, T (2009): SOA Design Patterns, Prentice Hall, Upper Saddle River.
- (8) Franczyk, B (2010): Logistik-Service-Bus - Projekt. <http://www.lsb-plattform.de>. Abgerufen am 22.09.2011.
- (9) Gallas, B (2007): Enterprise Service Integration (ESI) - Der Weg zur einem servicebasierten EAI-Framework unter Einsatz und Erweiterung von Web Services. In: Aier, S; Schönherr, M. (Hrsg), Enterprise Application Integration - Flexibilisierung komplexer Unternehmensarchitekturen, Gito-Verl, Berlin.
- (10) Hohpe, G, Woolf, B, Brown, K (2010): Enterprise integration patterns: Designing, building, and deploying messaging solutions, Addison-Wesley, Boston.
- (11) Klinkmüller, C, et al (2011): The Logistics Service Engineering & Management Platform: Operations, Architecture, Implementation. In: 14th International Conference on Business Information Systems, Poznań, 2011.
- (12) Kohlborn, T, Korthaus, A, Rosemann, M (2009): Business and software service lifecycle management. In: IEEE International Conference on Enterprise Distributed Object Computing, Auckland, New Zealand.
- (13) Kunkel, R, et al (2011): Modellgetriebene Integration von Logistik-Informationssystemen in die LSEM-Plattform. 41. GI-Jahrestagung Informatik 2011, Die Rolle von Plattformen für Unternehmensökosysteme, Berlin, 2011.
- (14) Luo, Z (2010): Service science and logistics informatics. Innovative perspectives. Premier reference source, Business Science Reference, Hershey Pa.
- (15) Petkoff, B (2007): Model Driven Application Integration am Beispiel der Versicherungswirtschaft. In: Aier, S; Schönherr, M. (Hrsg.), Enterprise Application Integration - Flexibilisierung komplexer Unternehmensarchitekturen, Gito-Verl, Berlin.
- (16) Shneiderman, B (1997): Designing the User Interface: Strategies for Effective Human-Computer Interaction, Addison-Wesley Longman Publishing Co., Boston.
- (17) Stahl, T, et.al. (2007): Modellgetriebene Softwareentwicklung. Techniken, Engineering, Management. dpunkt Verlag, Heidelberg.
- (18) Thränert, M, Kühne, S (2008): Model-Driven Integration Engineering zur Integration betrieblicher Anwendungssysteme. In: Fähnrich, K; Kühne, S.; Thränert, M. (Hrsg.), Model-Driven Integration Engineering. Universität Leipzig Pressestelle, Leipzig.
- (19) Weerawarana, S, Meredith, G, Curbera, F, Christensen, E (2001): Web Services Description Language (WSDL) 1.1. <http://www.w3.org/TR/wsdl>. Abgerufen am 22.09.2011.

## **Vielen Dank für die Aufmerksamkeit!**

Robert Kunkel – Universität Leipzig  
Grimmaische Str. 12 – 04109 Leipzig  
I: <http://www.iwi.uni-leipzig.de>  
T: +49 (0)341 - 97 33 746  
F: +49 (0)341 - 97 33 729  
M: [kunkel@wifa.uni-leipzig.de](mailto:kunkel@wifa.uni-leipzig.de)