

## Ein modellgetriebener Ansatz zur Nutzung von WS-BPEL für Scientific Workflows

*Guido Scherp*

OFFIS – Institut für Informatik  
guido.scherp@offis.de

Wilhelm Hasselbring  
Universität Kiel  
wha@informatik.uni-kiel.de

Grid Workflow Workshop 2010  
Paderborn  
23. Februar 2010

# Grundlagen

## Scientific Workflows

### Fachliche Ebene

*„The main goals of scientific workflows, then, are (i) to save 'human cycles' by enabling scientists to focus on domain-specific (science) aspects of their work, rather than dealing with complex data management and software issues; and (ii) to save machine cycles by optimizing workflow execution on available resources.“*

[Ludäscher et al. 2009]

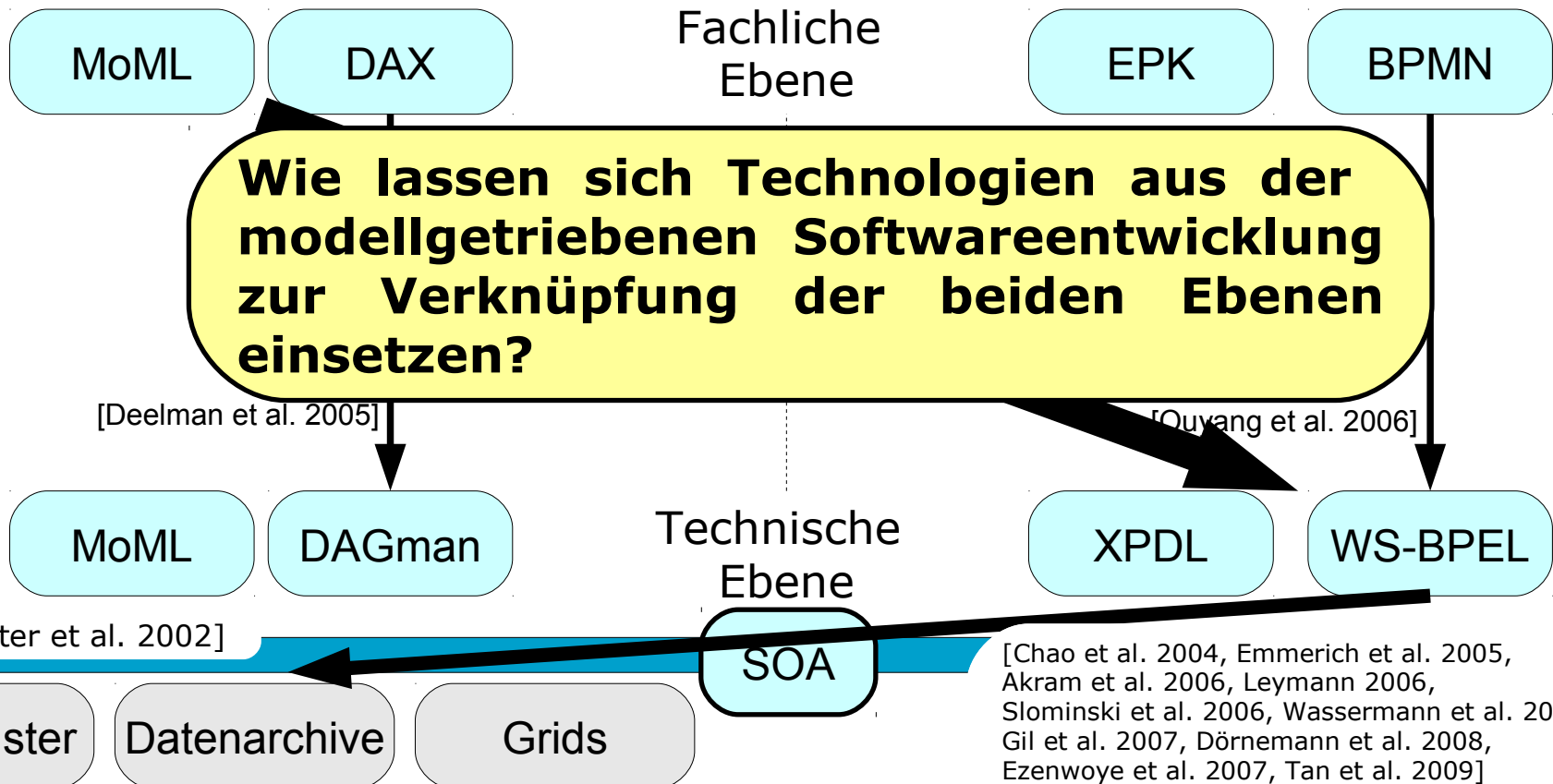
### Technische Ebene

# Motivation (1/2)

## Business Workflow-Technologien für Scientific Workflows

### Scientific Workflows

### Business Workflows



## Motivation (2/2)

### WS-BPEL für Scientific Workflows

- ▶ **OASIS-Standard**
- ▶ **Breite Unterstützung seitens der Industrie**
- ▶ **Zahlreiche Workflow-Engines verfügbar**
- ▶ **Als Web Service-Standard konzipiert - Nutzung Service-orientierter Grids**

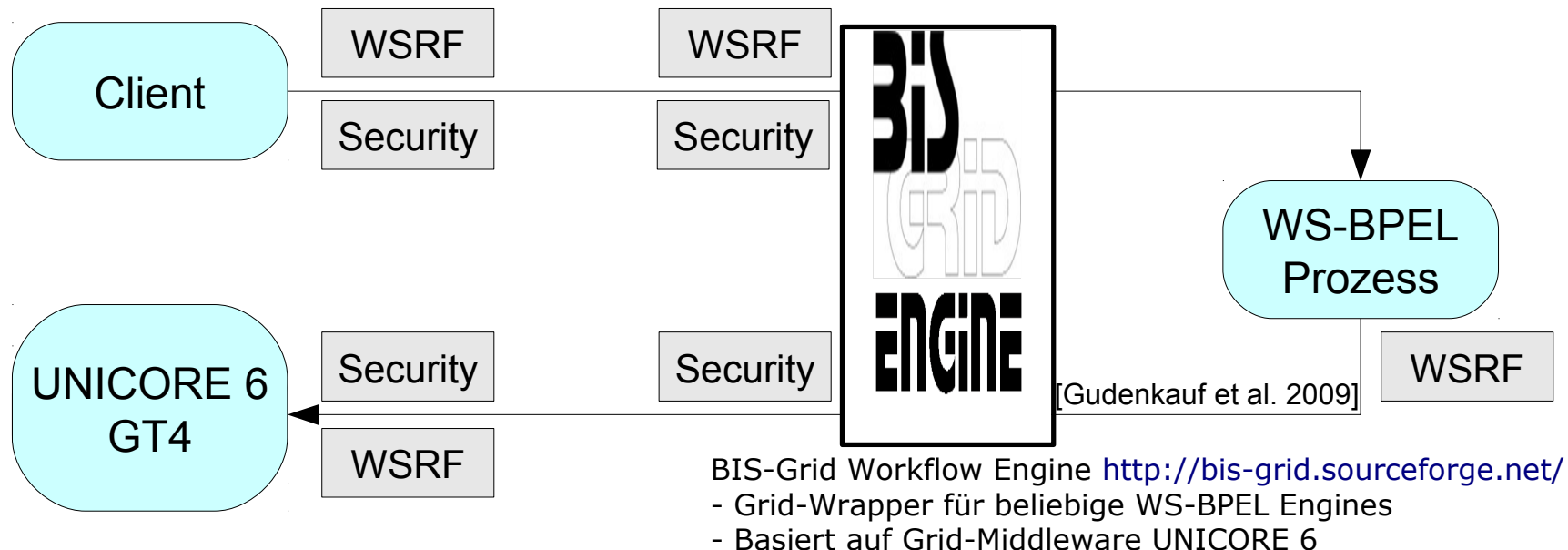
**WS-BPEL eignet sich *grundsätzlich* für die technische Ausführung von Scientific Workflows in Service-orientierten Grids**

# Ansatz (1/3)

## Scientific Workflows in Service-orientierten Grids

### Zu berücksichtigende Technologien

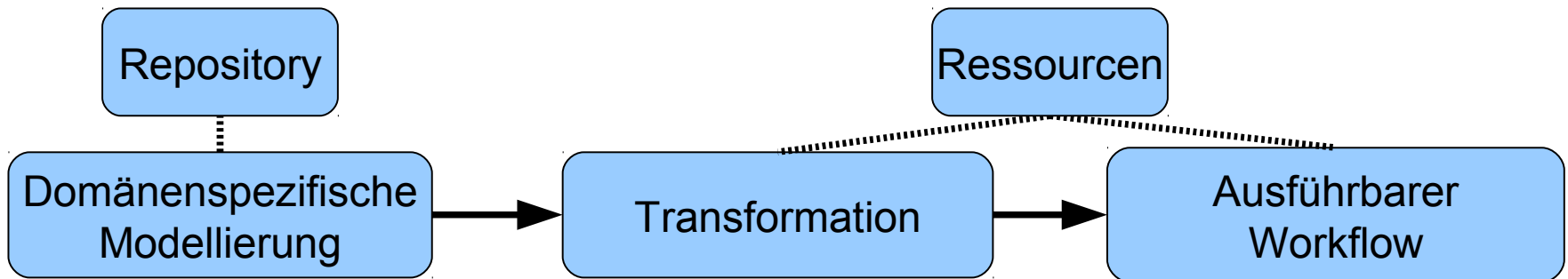
- ▶ **WSRF**
  - ▶ Aufrufen eines WSRF Web Services
  - ▶ WS-BPEL Prozess als WSRF Web Service anbieten
- ▶ **Sicherheitsstandards**



**Nur Betrachtung von WSRF Web Service-Aufrufen**

## Ansatz (2/3)

### Überblick



- ▶ **Elemente des Repository enthalten Informationen über verfügbare Datenquellen- und senken, Prozessierungen, usw.**
  - ▶ Dies beinhaltet auch technische Informationen zur Abbildung auf Ressourcen
- ▶ **Nutzer verwendet das Repository zur grafischen, domänenspezifischen Modellierung**
  - ▶ Technische Informationen für Nutzer sind (fast) unsichtbar
  - ▶ Datenfluss-orientiert
- ▶ **Transformation in einen ausführbaren Workflows auf Basis einer Business Workflow-Sprache wie WS-BPEL**
  - ▶ Statische oder dynamische Abbildung auf Ressourcen
  - ▶ Kontrollfluss-orientiert

## Ansatz (3/3)

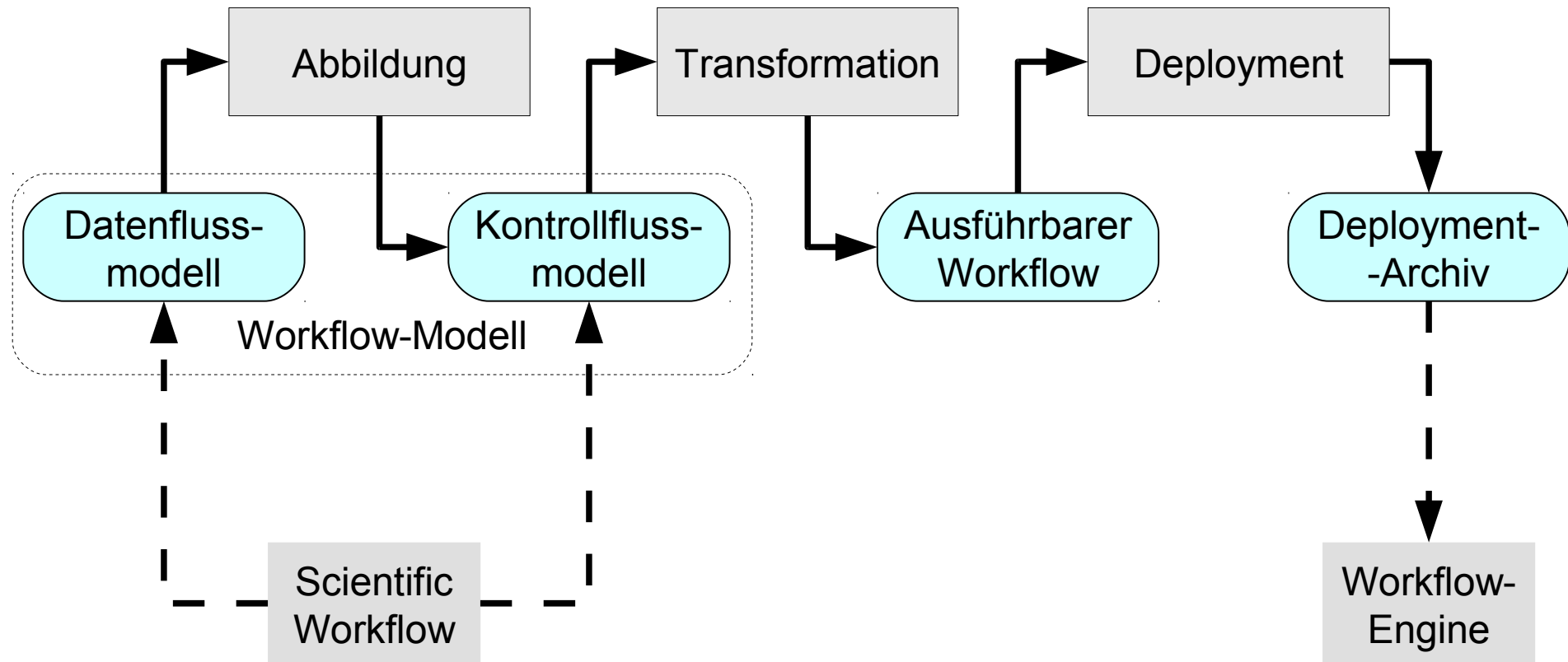
### Ziele

- ▶ **Nutzung etablierter Scientific Workflow-Technologien zur Modellierung von Scientific Workflows**
  - ▶ Keine Erfindung einer neuen „Sprache“
- ▶ **Nutzung etablierter und standardisierter Business Workflow-Technologien wie WS-BPEL zur Ausführung von Scientific Workflows**
  - ▶ Keine Erzeugung von Sprachdialekten
  - ▶ Fokus auf Service-orientierte Grid-Infrastrukturen
- ▶ **Nutzung von etablierten Technologien aus der modellgetriebenen Softwareentwicklung**
  - ▶ Automatische Generierung von komplexen Sprachkonstrukten

**Bereitstellung eines Transformations-Framework zur Generierung eines ausführbaren Scientific Workflows auf Basis einer Business Workflow-Sprache wie WS-BPEL**

# Transformations-Framework (1/4)

## Überblick

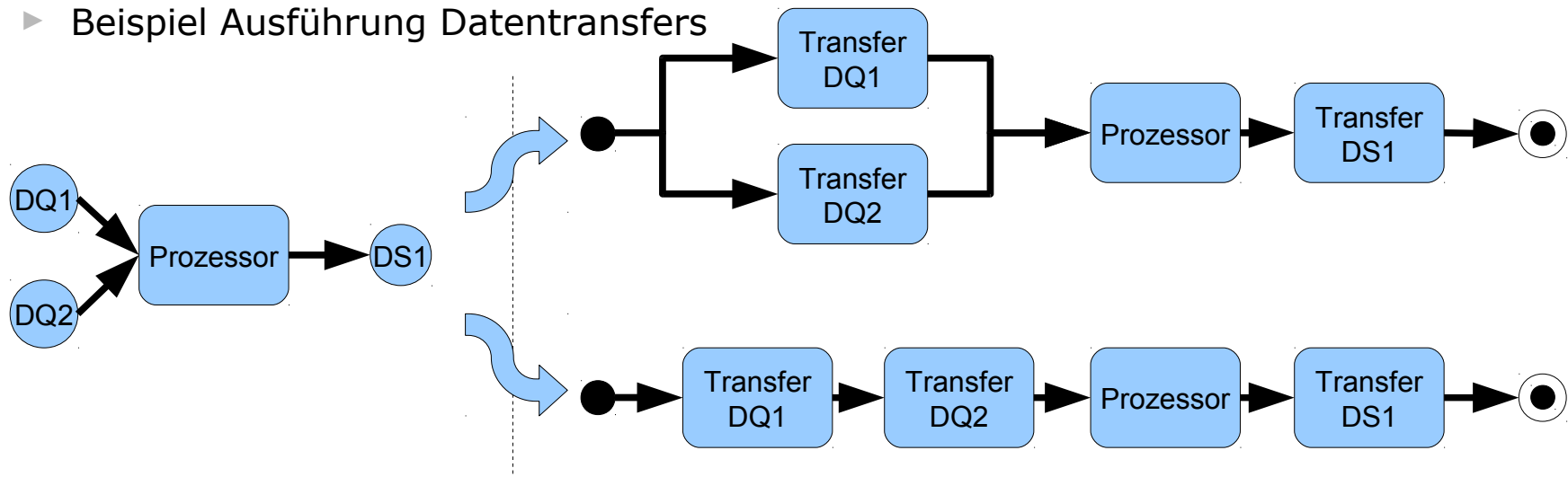




# Transformations-Framework (2/4)

## Darstellung von Datenfluss- und Kontrollfluss / Abbildung

- ▶ **Intermediäres Workflow-Modell zur Darstellung von Daten- und Kontrollfluss**
  - ▶ Unabhängigkeit von der (fachlichen) Scientific Workflow-Beschreibung
- ▶ **Übertragung von Daten- und Kontrollflussinformationen in das intermediäre Workflow-Modell**
  - ▶ Datenfluss-orientierte Sicht führt zu „fehlenden Kontrollflussinformationen“
- ▶ **Abbildung „fehlender Kontrollflussinformationen“**
  - ▶ Beispiel Ausführung Datentransfers



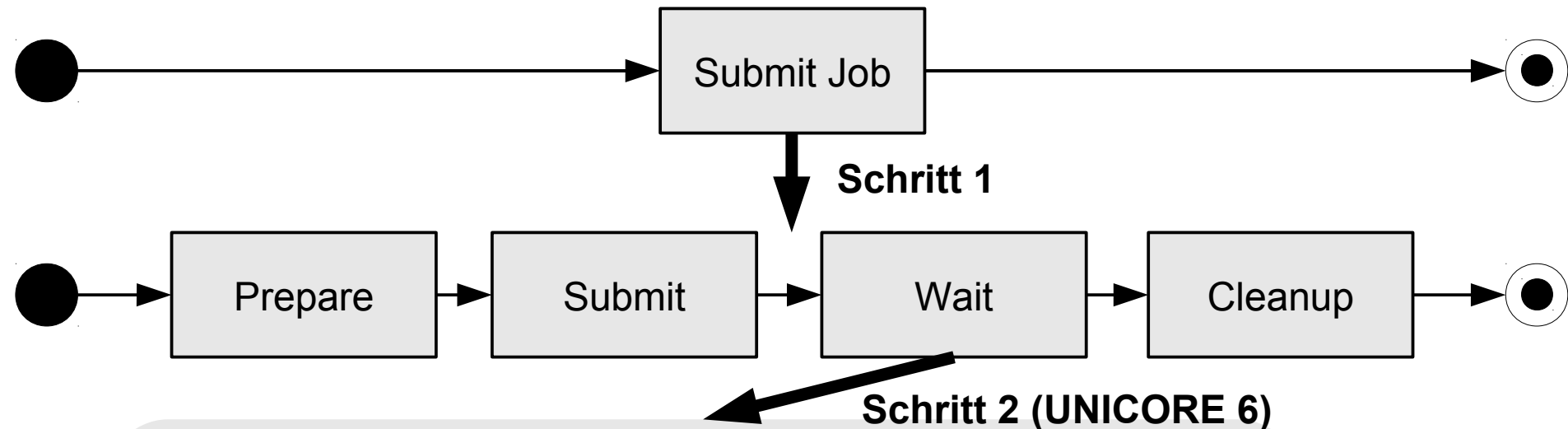
# Transformations-Framework (3/4)

## Transformation

- ▶ **Transformation des Workflows in die Zielspache**
  - ▶ Transformation ausgehend von Kontrollfluss-orientierter Sicht des Workflow-Modells
  - ▶ Generierung von zusätzlichen Konstrukten wie Fehlerbehandlung, Überwachung, Anbindung externer Scheduler, usw.
  - ▶ Transformationsregeln sind strukturiert zur Erleichterung von Anpassungen und Erweiterungen
  
- ▶ **Schritt 1: Ersetzen von Aktivitäten durch typische Ablaufmuster**
  - ▶ Z. B. *Job-Submission* durch *Prepare, Submit, Wait, Cleanup*
  - ▶ Unabhängig von der Infrastruktur (z. B. UNICORE 6 oder Globus Toolkit 4)
  
- ▶ **Schritt 2: Erzeugung des Quelltextes in der Zielsprache**
  - ▶ An die Infrastruktur angepasst
  - ▶ Z. B. UNICORE 6 und Globus Toolkit 4

# Transformations-Framework (4/4)

## Beispiel Job-Submission



```

<bpel:repeatUntil name="wait_until_job_done"
  <bpel:condition>
  $GetResourcePropertyDocumentResponse...../typ:Status/text() != 'RUNNING' and
  $GetResourcePropertyDocumentResponse...../typ:Status/text() != 'QUEUED' and
  $GetResourcePropertyDocumentResponse...../typ:StatusInfo/typ:Status/text() != 'READY'
</bpel:condition>
  <bpel:sequence name="check_job_state">
    <bpel:invoke inputVariable="GetResourcePropertyDocumentRequest"
      name="GetJMS_RP" operation="GetResourcePropertyDocument"
      outputVariable="GetResourcePropertyDocumentResponse"
      partnerLink="JobSubmissionPLT"
      portType="rpw-2:GetResourcePropertyDocument"/>
    <bpel:wait name="Wait10sec">
      <bpel:for>'PT10S'</bpel:for>
    </bpel:wait>
  </bpel:sequence>
</bpel:repeatUntil>
  
```

# Implementierung und Fallstudie

- ▶ **Aktuell: Darstellung von Kontroll- und Datenfluss**
  - ▶ Womit beschreibe ich Kontroll- und Datenfluss?
  - ▶ Zwei getrennte Modelle – Ein „hybrides“ Modell?
  - ▶ Einschränkungen von Kontroll- und Datenfluss (Workflow Patterns, Graph-basiert vs. Block-orientiert, ...)?
- ▶ **Danach: Prototypische Implementierung des Transformations-Frameworks**
  - ▶ Nutzung von Ecore-Technologien des EMF
  - ▶ Ecore-Modelle für intermediäres Workflow-Modell und ausführbaren Workflow
- ▶ **Geplante Fallstudie**
  - ▶ Erweiterung der GT4-Testszenarios (mit GDI-Grid) in BIS-Grid

## Verwandte Arbeiten

- ▶ **Zahlreiche bestehende Scientific Workflow-Technologien wie GWES, Kepler, Pegasus, Taverna, Triana, usw.**
  - ▶ Kein Ersatz dieser Technologien
  - ▶ Ansatz zur Erweiterung zur Nutzung von Service-orientierten Grids auf Basis einer Business Workflow-Sprache wie WS-BPEL
  
- ▶ **WS-BPEL für Scientific Workflows** [Chao et al. 2004, Emmerich et al. 2005, Akram et al. 2006, Leymann 2006, Slominski et al. 2006, Dörnemann et al. 2007, Ezenwoye et al. 2007, Tan et al. 2009]
  - ▶ Eignung von WS-BPEL zur Ausführung von Scientific Workflows in Service-orientierten Grids gezeigt
  - ▶ Teilweise Erzeugung von Sprachdialekten
  - ▶ Werkzeuge bieten geringe technische Abstraktion
  
- ▶ **Abbildung fachliche → technische Ebene** [Deelman et al. 2005, Wassermann et al. 2006]
  - ▶ Fokussierung auf eine Workflow-Sprache
  - ▶ Keine Nutzung von modellgetriebenen Technologien zur Transformation

## Literatur (1/2)

- ▶ **[Foster et al. 2002]** I. Foster, C. Kesselman, J. M. Nick, S. Tuecke: „*The Physiology of the Grid An Open Grid Services Architecture for Distributed Systems Integration*“, 2002
- ▶ **[Chao et al. 2004]** K.-M. Chao, M. Younas, N. Griths, I. Awan, R. Anane, C.-F. Tsai: „*Analysis of Grid Service Composition with BPEL4WS*“, IEEE, 2004
- ▶ **[Emmerich et al. 2005]** W. Emmerich, B. Butchard, L. Chen, S. L. Price, B. Wassermann: „*Grid Service Orchestration Using the Business Process Execution*“, Springer, 2005
- ▶ **[Deelman et al. 2005]** Ewa Deelman, Gurmeet Singh, Mei-Hui Su, James Blythe, Yolanda Gil, Carl Kesselman, Gaurang Mehtaa, Karan Vahia, G. Bruce Berriman, John Goodb, Anastasia Laity, Joseph C. Jacob and Daniel S. Katz: „*Pegasus: A framework for mapping complex scientific workflows onto distributed systems*“, Scientific Programming Journal, 2005
- ▶ **[Yu et al. 2005]** J. Yu , R. Buyya: „*A Taxonomy of Scientific Workflow Systems for Grid Computing*“, ACM Press, 2005
- ▶ **[Akram et al. 2006]** A. Akram, D. Meredith, R. Allan: „*Evaluation of BPEL to Scientific Workflows*“, 2006
- ▶ **[Leymann 2006]** F. Leymann: „*Choreography for the Grid: towards fitting BPEL to the resource framework*“, 2006
- ▶ **[Ouyang et al. 2006]** Chun Ouyang , Marlon Dumas, Arthur H. M. ter Hofstede, Wil M. P van der Aalst: „*From BPMN Process Models to BPEL Web Services*“, ICWS '06, IEEE, 2006
- ▶ **[Slominski et al. 2006]** A. Slominski: „*On using BPEL extensibility to implement OGSI and WSRF Grid work*“, 2006
- ▶ **[Wassermann et al. 2006]**: B. Wassermann, W. Emmerich, B. Butchart, N. Cameron, L. Chen, J. Patel: „*Sedna: A BPEL-Based Environment for Visual Scientific Workflow Modeling*“, 2006
- ▶ **[Barga et al. 2007]** R. Barga, D. Gannon: „*Scientific versus Business Workflows*“, 2007
- ▶ **[Dörnemann et al. 2007]** T. Dörnemann, T. Friese, S. Herdt, E. Juhnke, B. Freisleben: „*Grid Workflow Modelling Using Grid-Specific BPEL Extensions*“, 2007

## Literatur (2/2)

- ▶ **[Ezenwoye et al. 2007]** O. Ezenwoye, S. M. Sadjadi, A. Cary, M. Robinson: „*WSRF-based Grid Services.*“, 2007
- ▶ **[Gil et al. 2007]** Y. Gil et al.: „*Examining the Challenges of Scientific Workflows*“, 2007
- ▶ **[Taylor et al. 2007]** I.J. Taylor, E. Deelman, D.B. Gannon, M. Shields: „*Workflows for e-Science: Scientific Workflows for Grids*“, Springer, 2007
- ▶ **[Yu et al. 2007]** Xiaofeng Yu, Yan Zhang, Tian Zhang, Linzhang Wang, Jianhua Zhao, Guoliang Zheng, Xuandong Li: „*Towards a Model Driven Approach to Automatic BPEL Generation*“, 2007
- ▶ **[Dörnemann et al. 2008]** Tim Dörnemann, Matthew Smith, Bernd Freisleben: „*Composition and Execution of Secure Workflows in WSRF-Grids*“, Cluster Computing and the Grid, IEEE, 2008
- ▶ **[Reussner, Hasselbring 2008]** Ralf Reussner, Wilhelm Hasselbring: „*Handbuch der Software-Architektur*“, dpunkt.verlag, 2008
- ▶ **[Ludäscher et al. 2009]** B. Ludäscher, M. Weske, T. McPhillips, S. Bowers: „*Scientific Workflows: Business as usual?*“, 2009
- ▶ **[Gudenkauf et al. 2009]** Stefan Gudenkauf, Wilhelm Hasselbring, Andre Höing, Guido Scherp, Odej Kao: „*Workflow Service Extensions for UNICORE 6 - Utilising a Standard WS-BPEL Engine for Grid Service Orchestration*“, 2009
- ▶ **[Scherp et al. 2009]** G. Scherp, A. Höing, S. Gudenkauf, W. Hasselbring, O. Kao: „*Using UNICORE and WS-BPEL for Scientific Workflow Execution in Grid Environments*“, 2009
- ▶ **[Tan et al. 2009]** W. Tan, P. Missier, R. Madduri, I. Foster. „*Building scientific workflow with Taverna and BPEL: A comparative study in caGrid*“, 2009

*Fragen?*



# *Zusatzfolien*

# Grundlagen

## Scientific Workflows vs. Business Workflows

	<i>Scientific Workflows</i>	<i>Business Workflows</i>
Domäne	Explorative Anwendungen; Prozessierung großer Datenmengen	Geschäftsprozesse; Betriebliche Informationsverarbeitung
Lebenszyklus	Eher explorative Workflows; häufige Änderungen	Eher produktive Workflows; wenig Änderungen
Modellierung	Datenfluss-orientiert	Kontrollfluss-orientiert
Datenverarbeitung	Dezentrale Prozessierung und Datentransformationen; Datenfluss über <i>third-party transfers</i>	Datentransformationen z. T. im Workflow modelliert; Datenfluss über Workflow-Engine
Rollenmodell	Eine Rolle und ein Workflow- Beteiligter	Mehrere Rollen und mehrere Workflow-Beteiligte

[Akram et al. 2006, Barga et al. 2007, Wassermann et al. 2007, Ludäscher et al. 2009, Scherp et al. 2009, Tan et al. 2009]

# Grundlagen

## Modellgetriebene Softwareentwicklung

