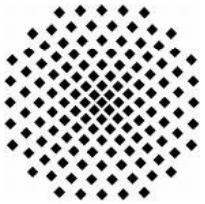
The Missing Features of Workflow Systems for Scientific Computations

Mirko Sonntag, Dimka Karastoyanova, Frank Leymann
{sonntag, karastoyanova, leymann}@iaas.uni-stuttgart.de
Institute of Architecture of Application Systems (IAAS)



Universität Stuttgart
Germany

SimTech Stuttgart

- Cluster of projects with subject simulation technology
  - > 72 PhD projects
  - Systems biology, engineering, mathematics, philosophy, …
- Research topic of IAAS
  - Workflow management system for simulations
  - Based on business workflow technology

Agenda

- Introduction
  - Business and scientific workflow management (WFM)
- Discussing missing features
  - Tool integration
  - Deployment
  - Execution
  - Monitoring
  - Flexibility
  - Provenance
- Conclusion and outlook

Business Workflow Management

- Support of business processes since the 90s
- Nowadays established and well-proven
- General purpose systems
  - Universally designed
  - Independent of concrete business area
- High complexity
  - Lots of configuration options
- Process = product
- Exemplary systems
  - Commercial: IBM WebSphere family, Oracle SOA Suite, …
  - Open Source: Apache ODE, Bexee, Freeflou, …

Scientific Workflow Management

- Young research field
- Used to support scientists
  - Workflows implement experiments, simulations, computations
- Usually built from scratch
- Tailored to particular domains
- Exemplary systems
  - Kepler, Taverna, Triana, Pegasus, SEGL, e-BioFlow

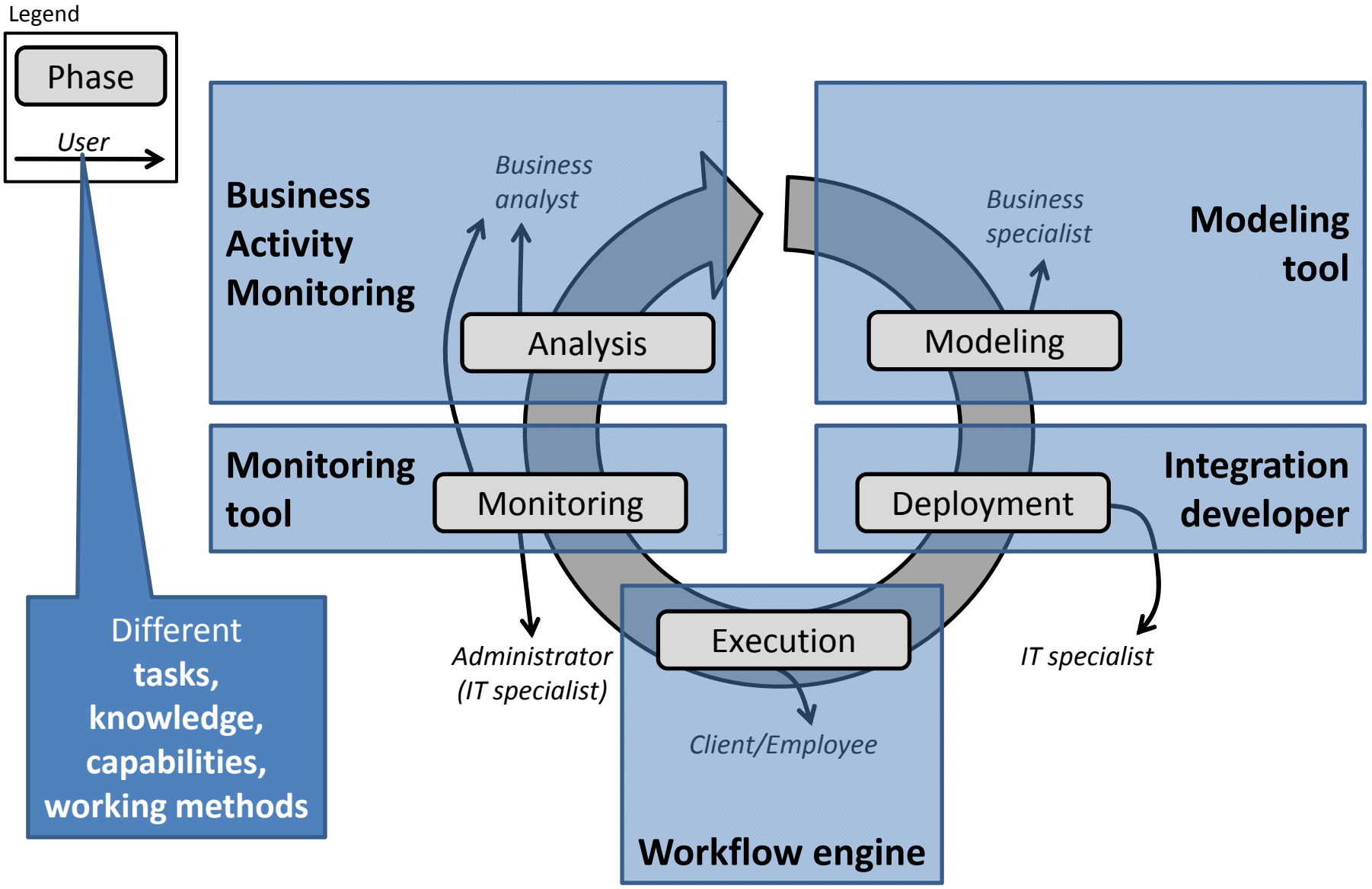
Business Workflow Technology for e-Science

- Endeavor to join the business and scientific WF community
  - i.e. to use business WfMSs in the scientific domain
- Advantages
  - Collaboration through use of standards
  - Existing tools as development basis
  - Robustness
  - Scalability (w.r.t. resources, workflow models, workflow instances)
  - …

Requirements on Scientific WfMSs

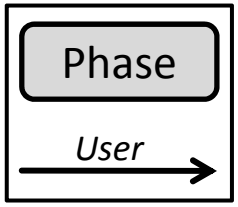
- Non-computer experts as users → high usability
- Long-running simulations
  - Can take hours, weeks or even months
- Importance of Data
  - Scientists think and model data-oriented
- Trial-and-error approach
  - Experiment design is an evolving process
- Grid support
  - To decrease run time of simulations + resource sharing
- Provenance
  - Reproducibility and trust

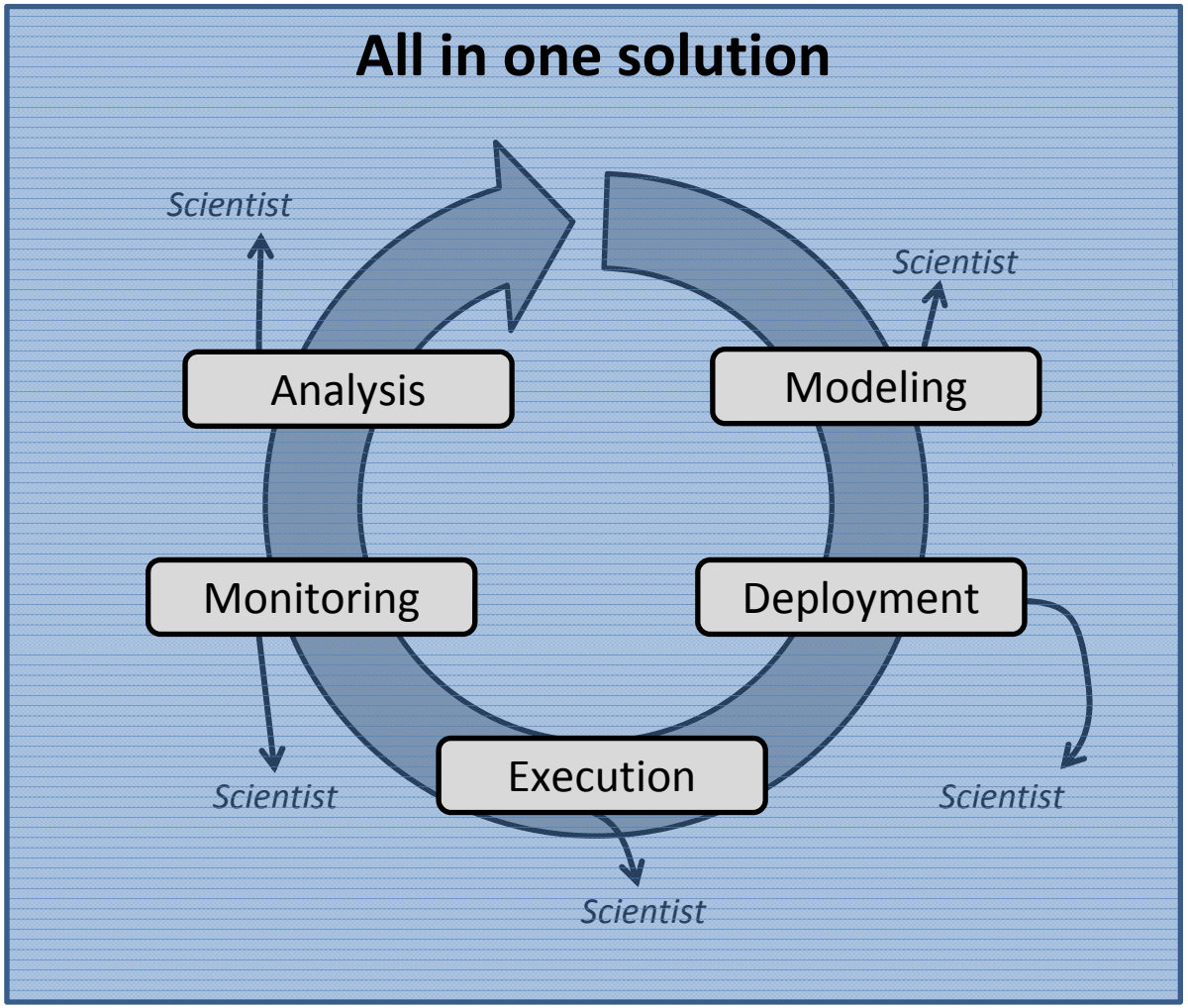
Life Cycle and Tool Integration



Life Cycle of Scientists

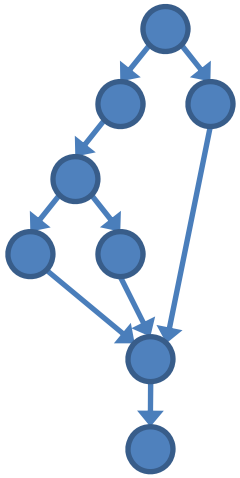
Legend





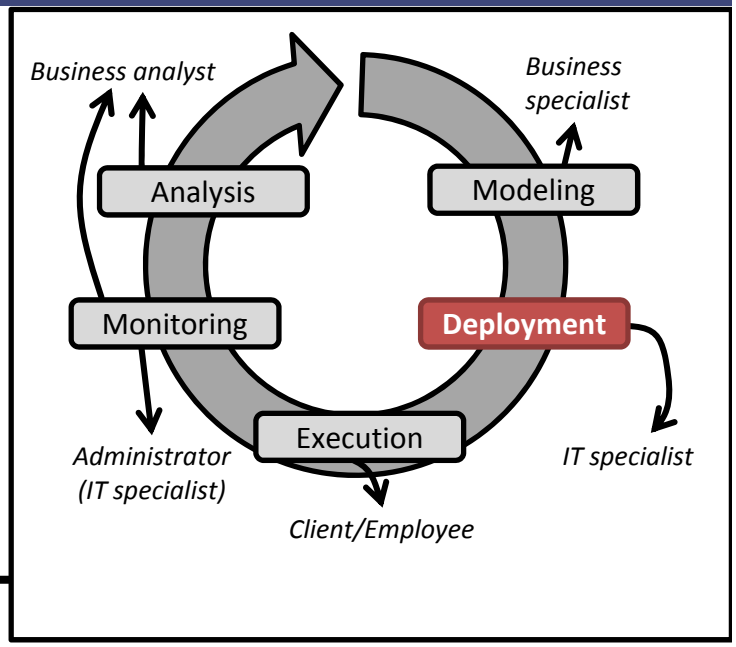
Deployment

Reuse of workflow parts



.dd

.bpel



Modeling

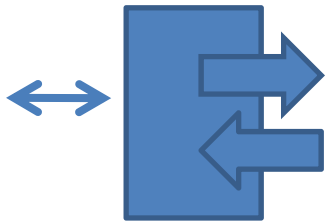
Execution

Efficient execution

Engine

Database

Robustness



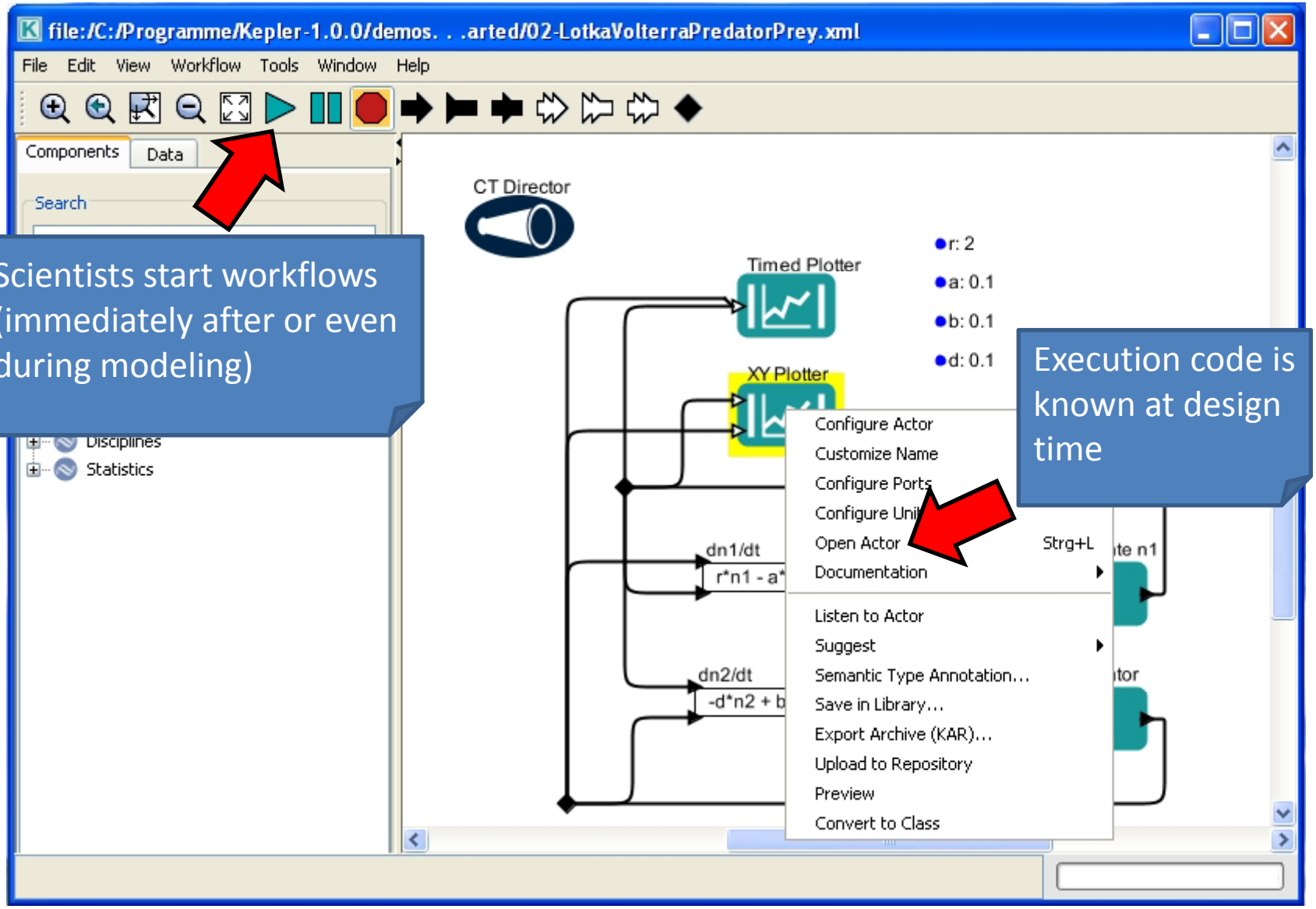
Web Service

Collaboration

Deployment Descriptor of Apache ODE

```
<deploy xmlns=„…" …>
  <process name="main:MagicSessionMain">
    <in-memory>true</in-memory>
    <active>true</active>
    <provide partnerLink="executePartnerLink">
      <service name="mws:MSMainExecuteService" port="MSExecutePort"/>
    </provide>
    <invoke partnerLink="responderPartnerLink">
      <service name="mws:MSResponderService" port="MSResponderPort"/>
    </invoke>
    <process-events generate="none">
      <enable-event>activityLifecycle</dd:enable-event>
      <scope-events name="aScope">
        <enable-event>dataHandling</bpel:enable-event>
        <enable-event>scopeHandling</bpel:enable-event>
      </scope-events>
    </process-events>
  </process>
</deploy>
```

Deployment in Scientific WFM – Kepler

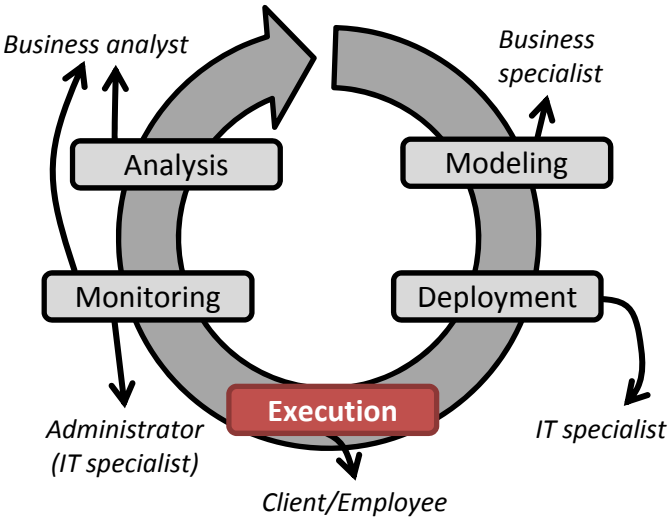


Deployment Specific Information in Kepler's MoML

```
<entity name="Open Database Connection" class="org.geon.OpenDBConnection">
  <property name="databaseFormat"
      class="ptolemy.data.expr.StringParameter" value="MySQL" />
  <property name="databaseURL"
      class="ptolemy.kernel.util.StringAttribute" value="jdbc:mySql://localhost:3306/" />
  <property name="username"
      class="ptolemy.data.expr.StringParameter" value="admin" />
  <property name="password"
      class="ptolemy.data.expr.StringParameter" value="password" />
</entity>
<entity name="Web Service Actor" class="org.sdm.spa.WebService">
  <property name="wsdlUrl" class="ptolemy.data.expr.StringParameter"
      value="http://localhost:8080/ode/testservice?wsdl" />
  <property name="methodName" class="ptolemy.data.expr.StringParameter"
      value="testOperation" />
  <property name="userName" class="ptolemy.data.expr.StringParameter"
      value="admin" />
  <property name="password" class="ptolemy.data.expr.StringParameter"
      value="password" />
</entity>
```

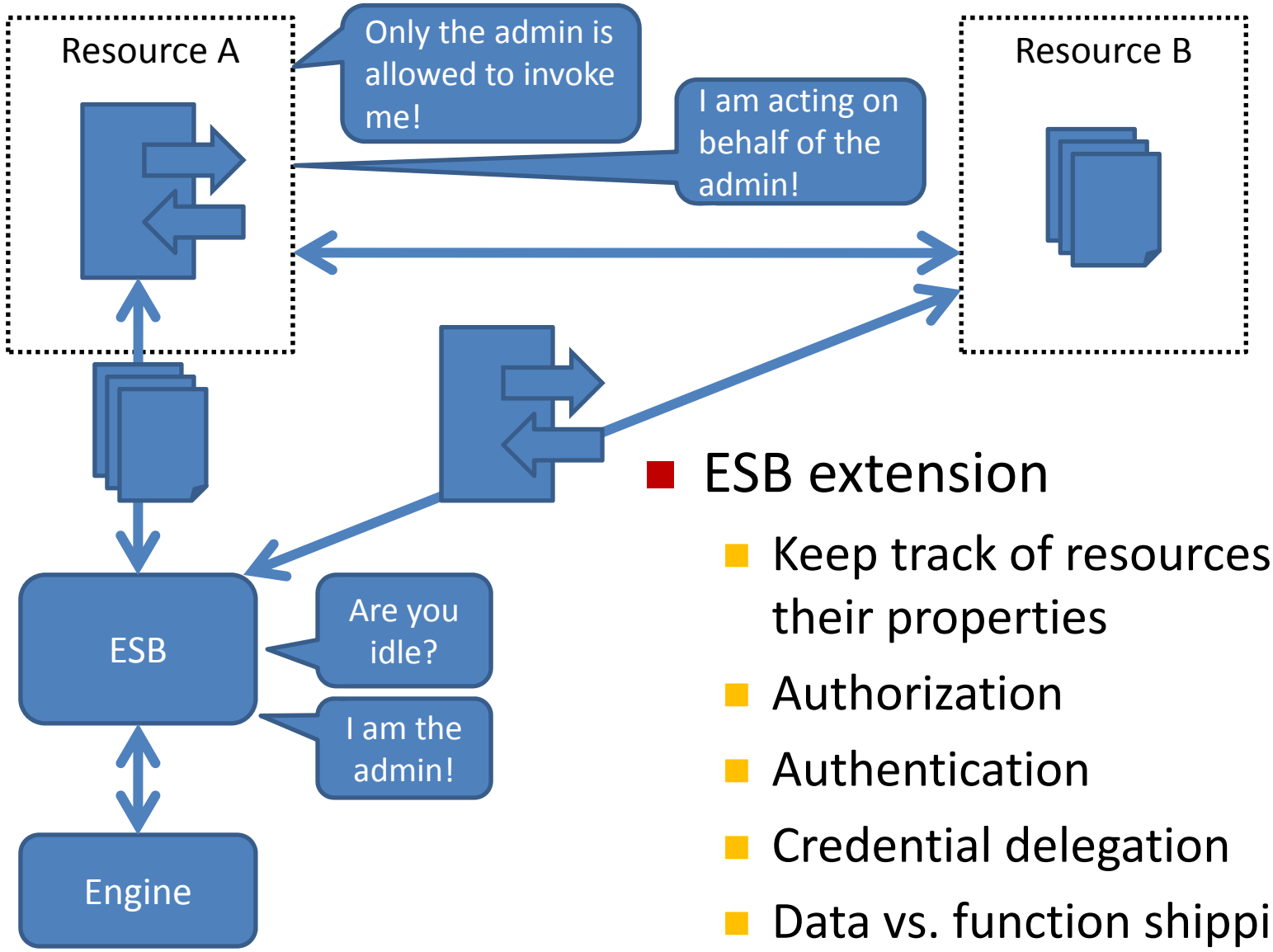
Execution

- Grid-awareness in business WfMSs via OGSA and WSRF
  - Resources can be accessed as stateful WSs



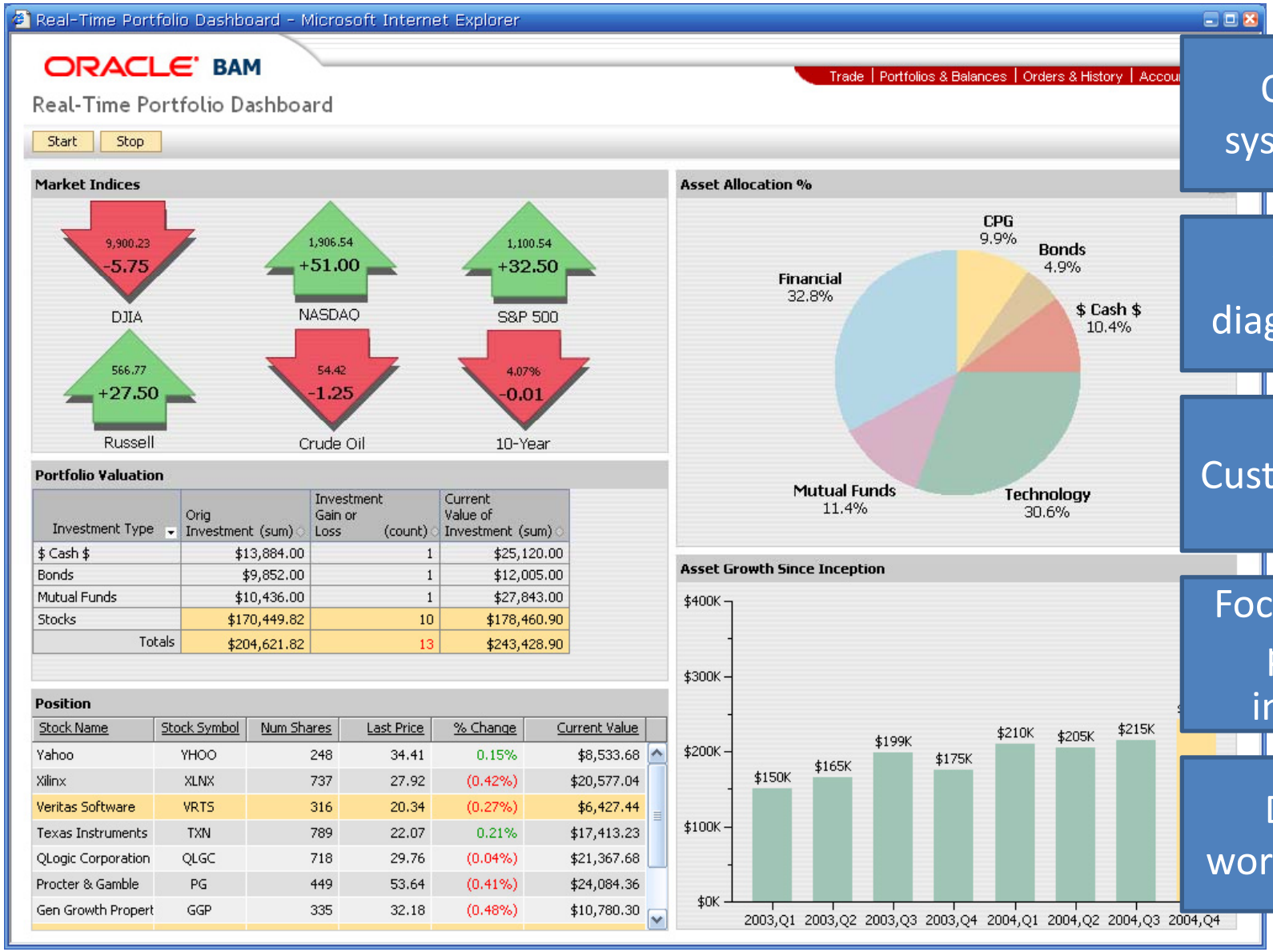
- Enterprise service bus (ESB) can be used to keep track of resources
  - Policy negotiation, finding, selecting, binding of services/resources

Recommendation for Execution



- ESB extension
  - Keep track of resources and their properties
  - Authorization
  - Authentication
  - Credential delegation
  - Data vs. function shipping

Monitoring & Analysis – Oracle BAM



Observe system state

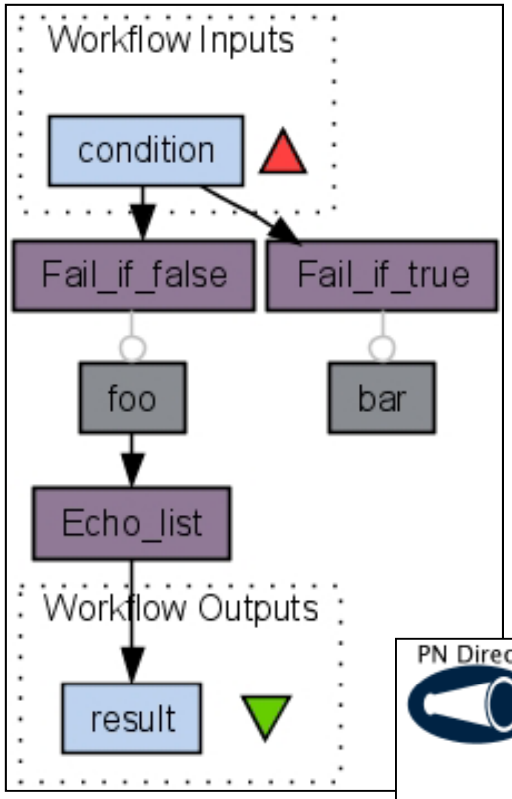
Tables, diagrams, etc.

Customizability

Focus: Lots of process instances

Data on workflow level

Monitoring and Analysis in Scientific WFM



Taverna

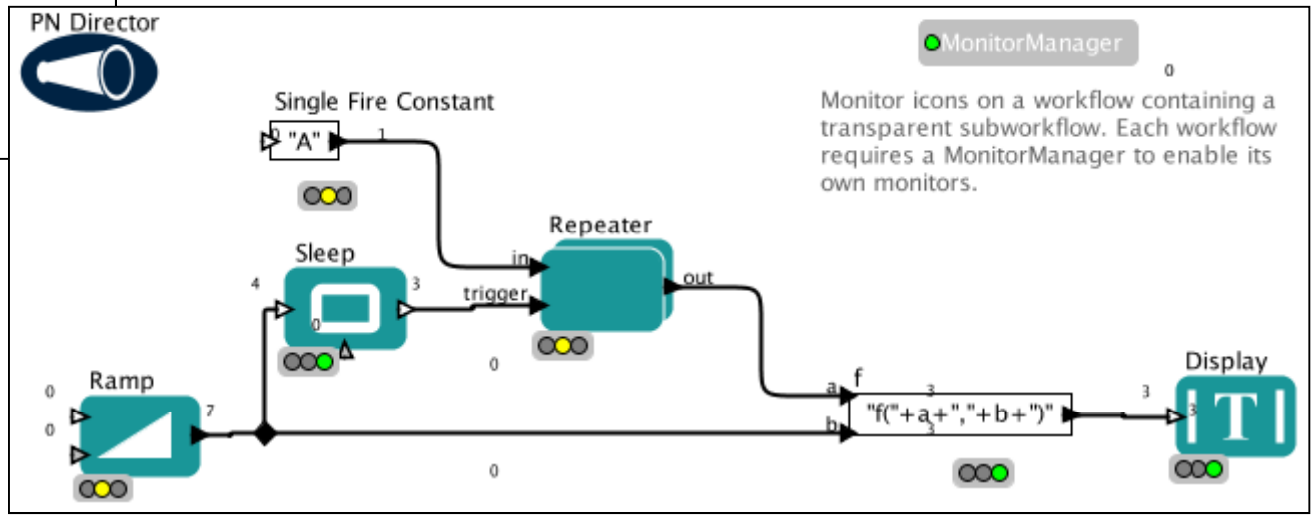
Scientists follow their experiments, simulations, …

Graph-oriented

Focus: single instances

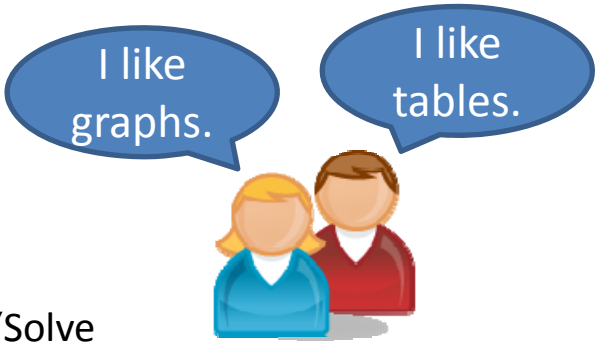
Restricted customizability

Kepler



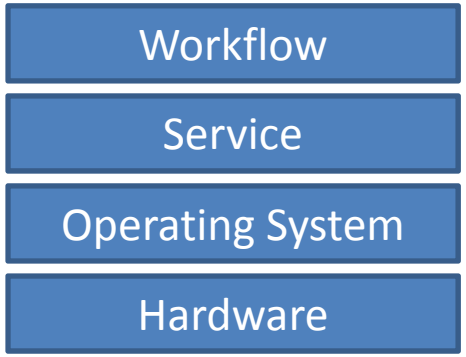
Recommendation for Monitoring and Analysis

- Adopt concepts
  - User notification
  - Customizability
  - Event model



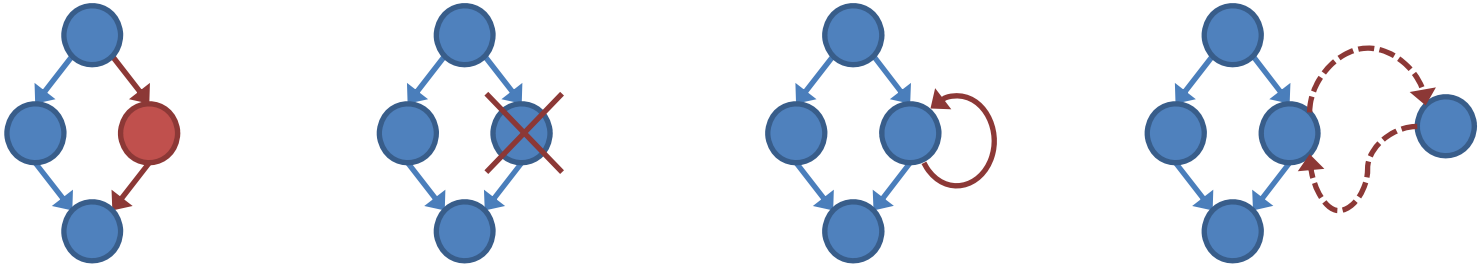
Activity “Solve equation system” has started

- Extensions needed
  - Focus on single instances
  - Monitoring in Grid environments



Flexibility in Business WFM

- Avoid change
- Workflow adaptation
  - Schema evolution
  - Ad hoc adaptation



Insert activity | Delete activity | Reiteration | Inquiry

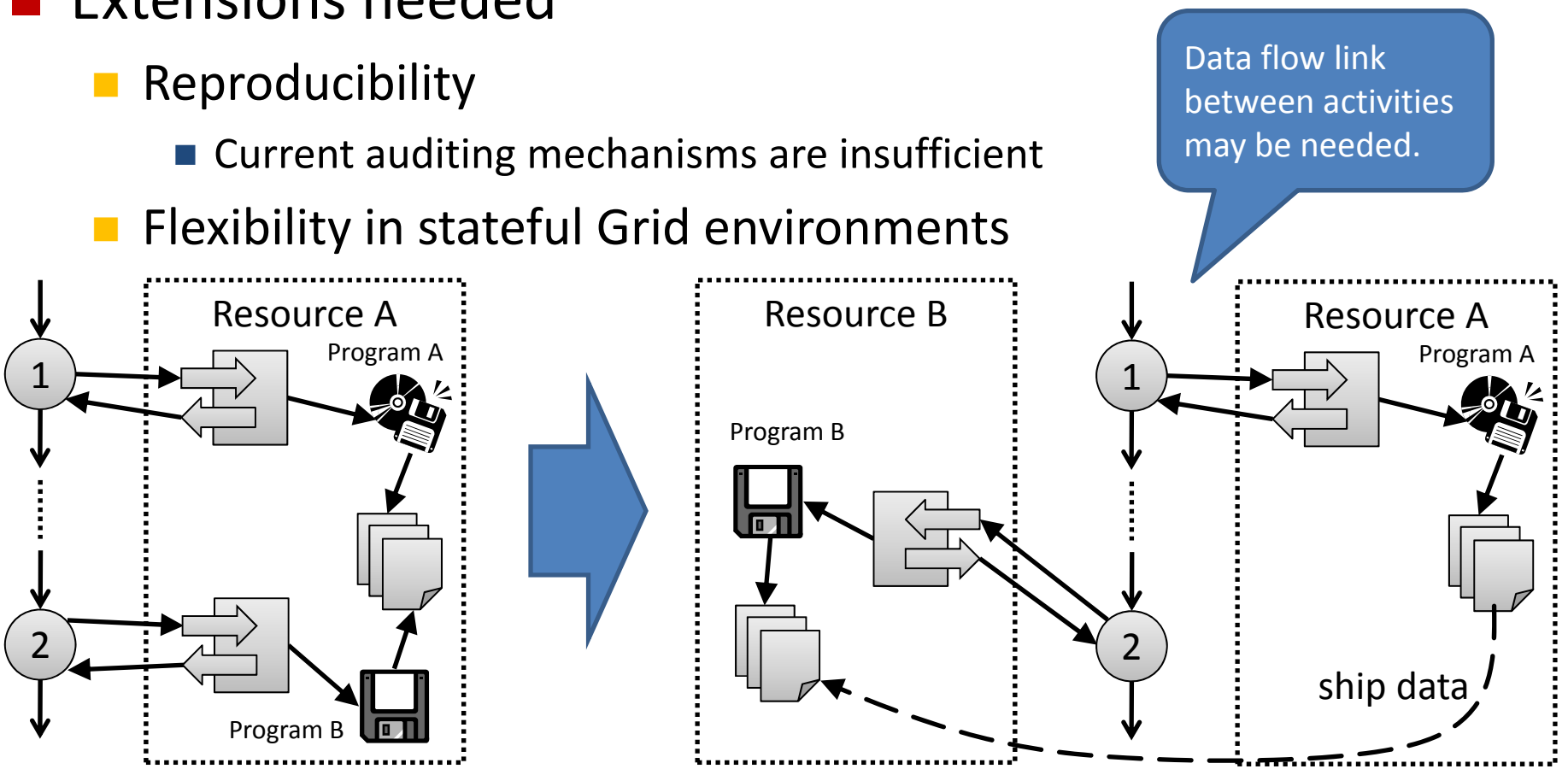
- Well-known, but insufficiently implemented
  - Versioning is typical approach

Flexibility in Scientific WFM

- Helps scientists to conduct experiments in a trial-and-error manner
- Currently almost unaddressed
  - Pegasus
    - Late binding of resources, workflow reduction, retry activities
  - Taverna
    - Retry activities, use alternative services
  - e-BioFlow
    - Re-execution of workflow fragments

Recommendation for Flexibility

- Adopt existing concepts from the workflow technology
  - Fortunately, instance migration is of minor importance
- Extensions needed
  - Reproducibility
    - Current auditing mechanisms are insufficient
  - Flexibility in stateful Grid environments



Provenance

- Guarantee reproducibility of results
- Current audit trails can contribute but are insufficient
  - On workflow level only
  - Information on tools, versions, operating systems, hardware needed
  - How were services found, selected, bound, invoked?
- ESB events are needed (at least)

Conclusion and Future Work

- There are reasons for the discrepancy between what business WfMSs deliver and what scientists need
- It is reasonable to apply the conventional WF technology to scientific applications
- But: we identified many missing features
- We provided recommendations how this can be done
  - And we know: this is hard work!
- Planned future work
  - Implement a scientific WfMS built on top of existing open-source WF engine, modeling tool, and service bus
  - Special attention on Grid awareness, monitoring, flexibility, data centrality

Questions ???

Thanks for your attention!
Any questions?