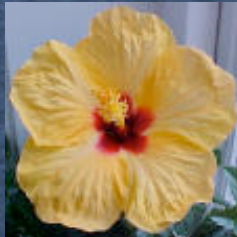


# SDSC Informnet: Workflow Description and Architecture

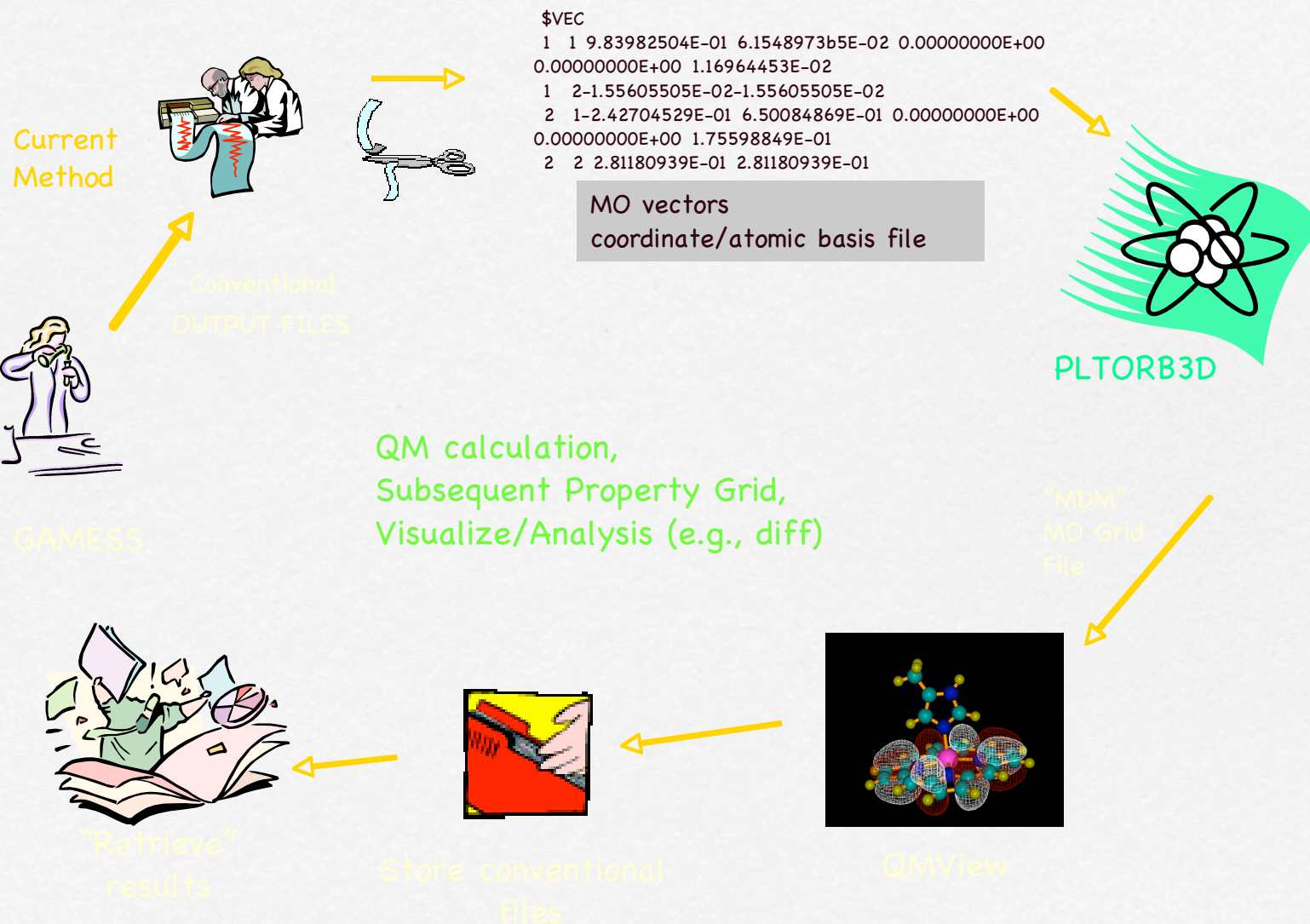
Stephen Mock  
San Diego Supercomputer Center  
GGF 11, O'ahu, Hawaii



# Portals to Workflows

- ❑ Collaboration with Quantum Chemistry group
  - ❑ Application or task oriented
  - ❑ Long running apps (days, weeks)
  - ❑ Smaller numbers of applications
  - ❑ High detail for inputs/outputs
- ❑ Desire higher degree of flexibility
  - ❑ Existing portals and flow languages lacked details of requirements

# Gameess Workflow Example



# Informnet Project

- Information Flow and Operation Resource Management on the NET
- PIs: Phil Papadopolous and Kim Baldrige
  - Jerry Greenberg, Stephen Mock
  - Backgrounds in Portals and Applications

# Outline

- ❑ **Informnet Goals and Motivation**
- ❑ **Workflow Description Language**
- ❑ **Architecture**

# Informnet Goals

- Late binding to computational resources
- Client detachability
- Grid aware
- Service based
- Support different client types (portals, java apps, etc)
- Interoperability with real world applications

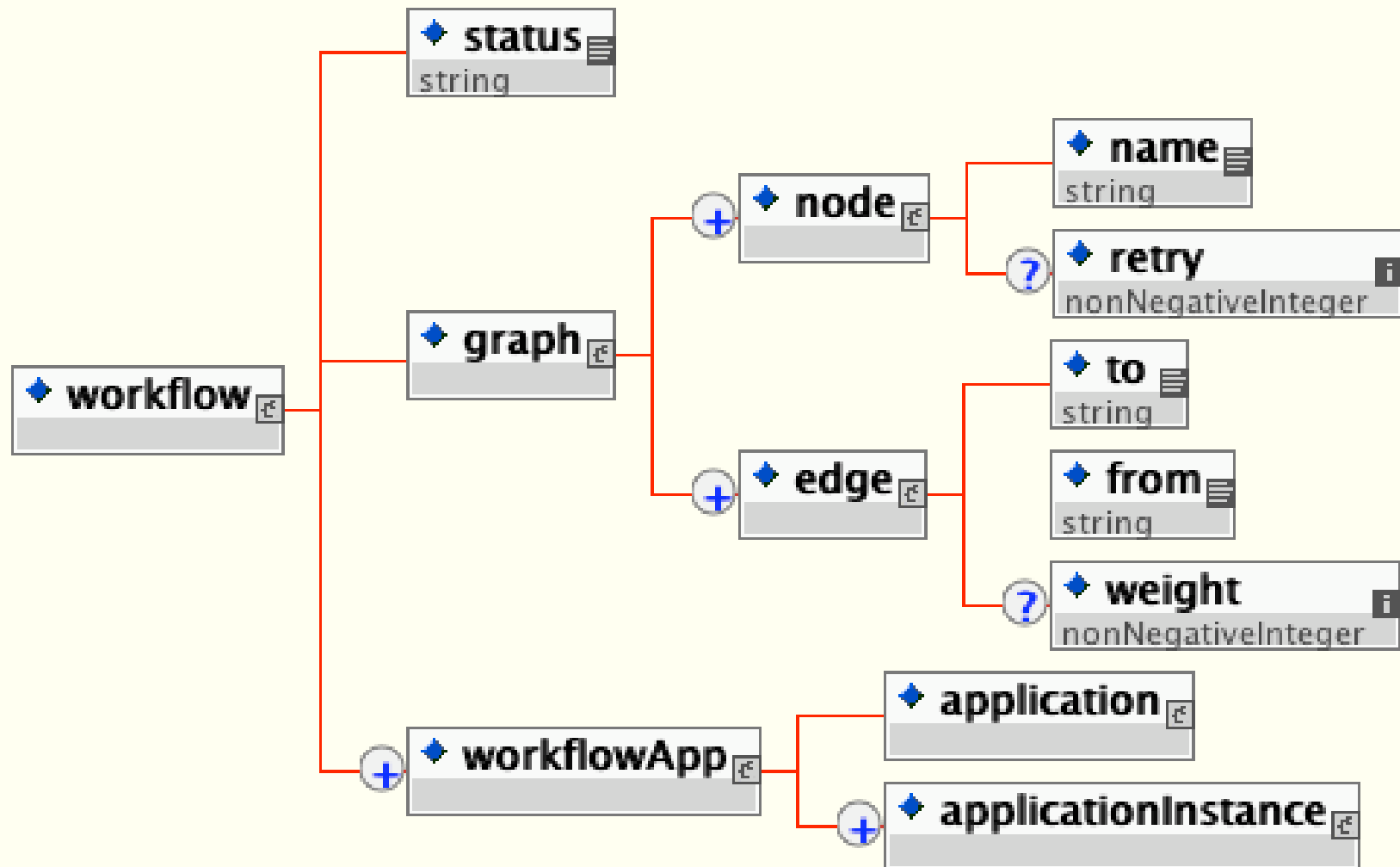
# Solutions

- ❑ Working with computational chemistry community to develop XML-based types
  - ❑ GAMESXML by Jerry Greenberg
  - ❑ CML by Peter Murray-Rust et al
- ❑ XML Workflow Description Language
- ❑ Grid Service Workflow Engine (GT3)

# Workflow Description

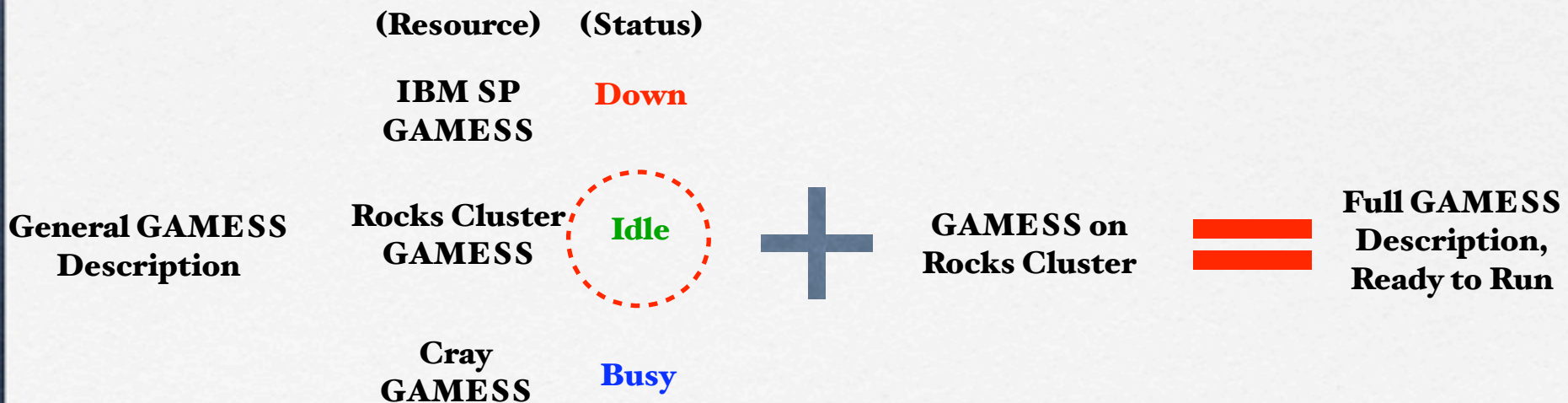
- XML Schema describes the workflow in two parts
  1. A Graph (DAG) for the order of execution
    - Nodes
    - Edges
  2. Description of the applications & requirements
    - High-level application details
    - Host-specific application details

# Workflow Schema



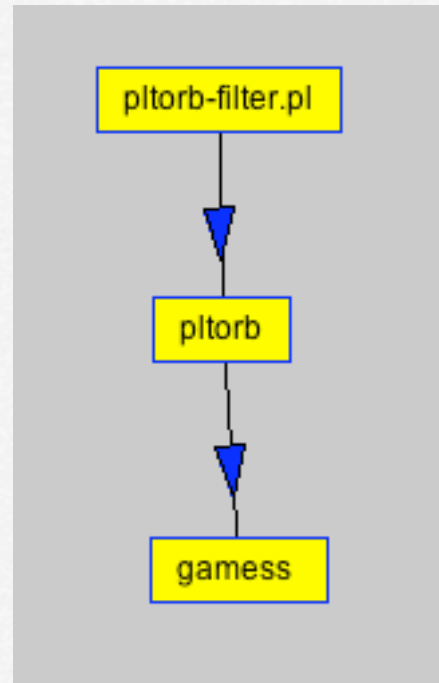
# 2 Application Types

- Separation of the general application information from resource specific details grants flexibility
  - Late binding to resources
  - Resource selection based on available, up to date, information
  - Brokering

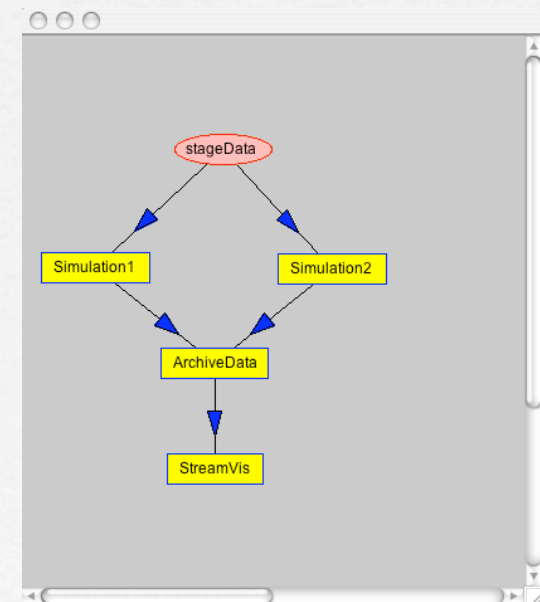


# Simple Example Graph XML

```
<graph>
  <node>
    <name>gameSS</name>
    <retry>1</retry>
  </node>
  <node>
    <name>pltorb-filter.pl</name>
    <retry>0</retry>
  </node>
  <node>
    <name>pltorb</name>
    <retry>1</retry>
  </node>
  <edge>
    <to>pltorb</to>
    <from>pltorb-filter.pl</from>
  </edge>
  <edge>
    <to>pltorb-filter.pl</to>
    <from>gameSS</from>
  </edge>
</graph>
```



*Slightly  
More Complex*

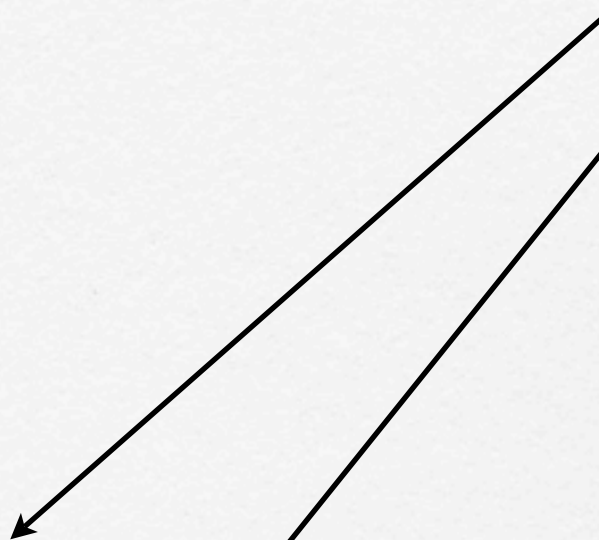


# General App Def.

```
<application name="gamess">
  <version>
    <major>2002</major>
    <minor>06</minor>
    <build>20</build>
    <suffix>beta</suffix>
  </version>
  <supportedArch>
    <archName>Intel-32</archName>
    <osName>Redhat Linux</osName>
    <osMajor>7</osMajor>
    <osMinor>1</osMinor>
    <cpu>Pentium 4</cpu>
    <compiler>
      <name>g77</name>
      <version>2.96</version>
      <language>fortran</language>
    </compiler>
  </supportedArch>
  <requirements>
    <simpleIOType>
      <name>gamess.inp</name>
      <polarity>INPUT</polarity>
      <transportType>local file</transportType>
      <description>gamess input file, plain ascii text</description>
    </simpleIOType>
    <simpleIOType>
      <name>gamess.dat</name>
      <polarity>OUTPUT</polarity>
      <transportType>local file</transportType>
      <description>gamess data output file, plain ascii text</description>
    </simpleIOType>
  </requirements>
</application>
```

1 Input File

1 Output File



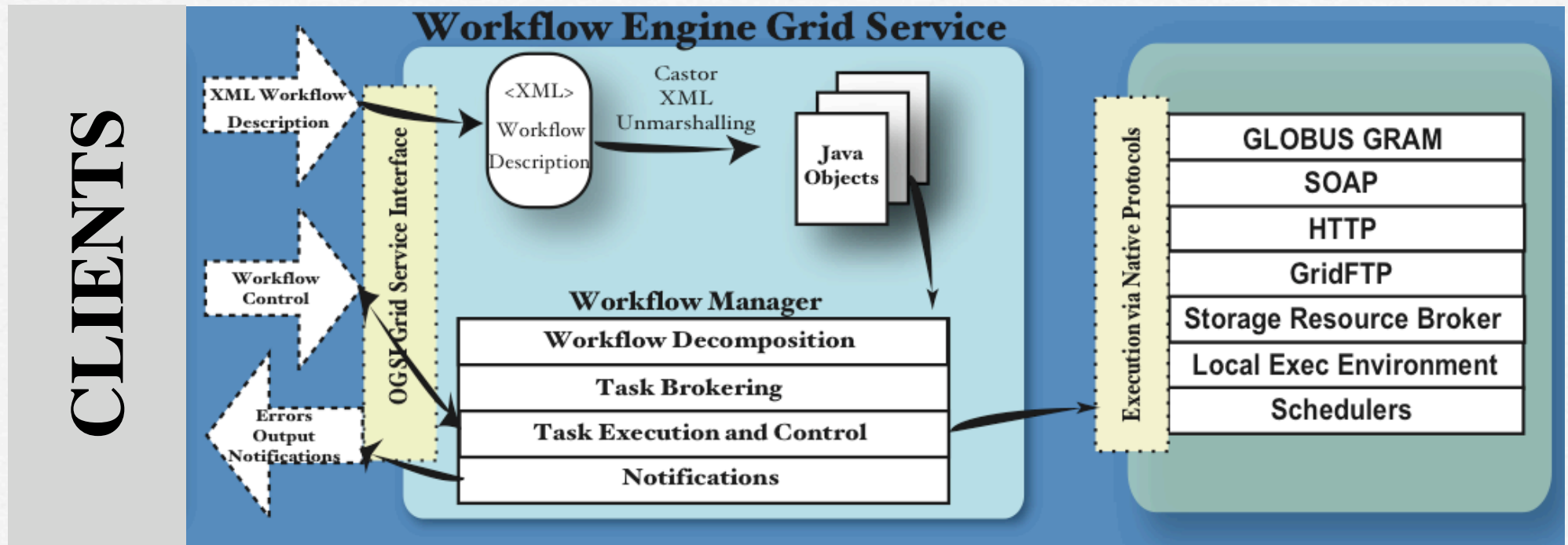
# Host-Specific App Def.

```
<applicationInstance>
  <resource>
    <name>grid-devel</name>
    <hostname>grid-devel.rocksclusters.org</hostname>
  </resource>
  <applicationInfo>
    <name>gamess</name>
    <executable>/usr/local/gamess/rungamessfile</executable>
    <arguments>/usr/local/gamess/tests/exam01 1 grid-devel.rocksclusters.org</arguments>
    <location>/usr/local/gamess</location>
    <version>20020620b</version>
    <applicationRequirements>
      <disk>100MB</disk>
      <memory>100MB</memory>
      <cpu>1</cpu>
      <scheduler>none</scheduler>
      <compiler>gcc</compiler>
      <IOTransport transportType="local file">
        <URI>
          <fileLabel>gamess.inp</fileLabel>
          <path>/usr/local/gamess/gamess.inp</path>
        </URI>
      </IOTransport>
      <IOTransport transportType="local file">
        <URI>
          <fileLabel>gamess.dat</fileLabel>
          <path>/home/mock/gamess.dat</path>
        </URI>
      </IOTransport>
    </applicationRequirements>
  </applicationInfo>
  <user>
    <userID>mock</userID>
    <authenticationType>gsi</authenticationType>
    <project>grid420</project>
  </user>
</applicationInstance>
```

1 Input File

1 Output File

# Workflow Engine Architecture



# GT3 Grid Service

- ❑ Workflow engine is a GT3 service
- ❑ exposes portTypes:
  - ❑ ingest workflow xml doc
  - ❑ workflow control and output gathering by client
- ❑ Notifications to client

# Summary

- **XML Workflow Description**
  - **Graph**
  - **General Application Definition**
  - **Host-specific Application Definition**
- **Grid Service Engine**

# Future

- **Future Directions**
  - **Looping, conditional branching**
  - **Support of JSDL job schema**
  - **Brokering policies**
  - **Ptolemy/Vergil Integration**
  - **GT4**

# Project Page

<http://chemport.sdsc.edu/informnet>

