Grid Scheduling Ontology

Proposed Working Group

Philipp Wieder, Research Center Jülich
Wolfgang Ziegler, Fraunhofer Institute SCAI

GLOBALGRIDFORUM.ORG

www.gridforum.org Grid Scheduling Ontology BoF, GGF10 Berlin, 10th March 2004

Agenda

Agenda

Motivation

Ontologies

Grid Scheduling Dictionary

Semantic Markup Languages & Tools

Grid Scheduling Ontology

GSO Working Group Charter

Contacts to other WGs/RGs

Next Steps

Contribute

Using rsl or similar for describing Grid Scheduling Requirements...

A user needs 16 nodes of a particular cluster:

```
+
( &(resourceManagerContact="packcs-e0/jobmanager=pbs")
   (count=16)
   (label="subjob 0")
   (environment=(GLOBUS_DUROC_SUBJOB_INDEX 0)
        (LD_LIBRARY_PATH /opt(globus/lib)
   (directory=/home/wolf)
   (executable=/home/wolf/plasim)
)
```

- User must have some grid middleware knowledge
- sysadmin has to manually re-do mapping for a new local scheduler
- rsl/scheduler interface has to be extended to support new resource attributes

... turns into a real challenge...

if a user requires slightly more than access to some compute nodes:

- A computing device with
  - 16 processing nodes,
  - operated under OS x
  - 1 GB of available memory per node, and
  - a licensed software package
  - for 1 hour between 8am and 6pm of the following day
- A local visualization device during program execution
- Minimum sustained bandwidth between the VR device and the computing device during program execution
- A specified data set from a data repository as input
- Should cost at most y €

  preference of cheaper job execution over an earlier execution.

... that might be accomplished through services only

Resource management and scheduling is a key service in a Next Generation Grid.

- In a large Grid the user cannot handle this task manually
- nor is finding and orchestration of resources a task of sysadmins

Many activities aiming to provide solutions to different aspects of the challenge (e.g. detection of resources, brokering, negotiation of SLAs) are under way: GRIP, myGrid, GESA, GRAAP, & many more

GSO - Grid Scheduling Ontology - will address the machine accessible semantics of the scheduling domain

What is an Ontology

There are many definitions of an Ontology.

Here are three we found useful for our work:

- An Ontology is a taxonomy combined with inference rules
- An Ontology is a formal explicit description of concepts in a domain of discourse
  - concepts or classes
  - properties of each concept describing features/attributes
  - restrictions
- Ontologies - the shared and common understanding of a domain
  - A vocabulary of terms
  - Definition what those terms mean
  - A shared understanding for people and machines

Some reasons why one would like to develop an Ontology

- To share common understanding of the structure of information among people or software agents

  allows software agents to extract and aggregate information from different sites, e.g. on different schedulers
- To enable reuse of domain knowledge

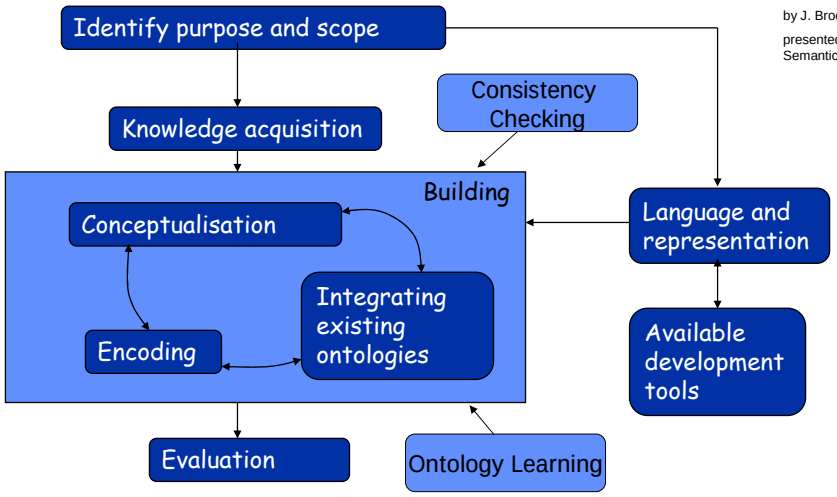
  existing Ontologies may be reused, e.g. an upper ontology of time, a resource ontology
- To make domain assumptions explicit

  easy changing of assumptions when knowledge changes, helpful for learning meaning of terms in a domain
- To separate domain knowledge from the operational knowledge

  e.g. a feeding an appropriate resource ontology to a scheduling service makes this service schedule computational resources, instruments, rooms...
- To analyse domain knowledge

  important issue when attempting to re-use or extend an existing ontology

Ontology building life-cycle



Semantic Matching of Grid Resource Description Frameworks

by J. Brooke, K. Garwood and D. Fellows

presented by Jon MacLaren at GGF9, Semantic Grid Workshopb

Grid Scheduling Dictionary Working Group

WG produced a document - GGF-I.11 - describing the Dictionary

2. List of Terms and Definitions

Term	Context	Definition
Account	Access	The permissions for a particular user to use various resources.
Authentication	Access	The process by which one entity verifies the identity of another.
User	Access	A person authorized to submit jobs to *High Performance Computing* resources.
Application Software	Application	The actual programs used by an application.
Checkpoint	Application	Saving the current state of a program and its data (including intermediate results) to disk or other non-volatile storage, so that if interrupted the program could be restarted at the point at which the last Checkpoint occurred. This is a feature provided by some schedulers, but not all.
...		
Start Time	Resource Management	A specification of the time when a job is expected to run or actually began running. May be specified as an actual time or as an offset from the current system time.
Time	Resource Management	Usually refers to *Execution Time* of a job.
...		
Usage	Resource Management	A measurement of the compute resources accessed by a user. This is typically a combination of *Nodes* and execution time.
Wallclock Time	Resource Management	The elapsed time between when a process starts to run and when it is finished. The Wallclock Time is normally longer than the *CPU Time* due to time-sharing *CPUs* and I/O.

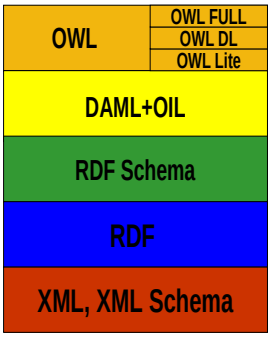
The Dictionary is an unstructured list of terms with only few links or references between them.

GSO WG will use Dictionary as base, adding/removing terms to re-focus

Turning it into an Ontology will generate classes with properties, relationships, and restrictions.

Semantic Markup Languages

- XML defines syntax rules for markup languages
- Semantic markup languages attempt to encode meaning within the markup language
- RDF is the simplest of the markup languages
- RDF Schema provides formal definitions of RDF and tools, adds useful classes and properties
- DARPA Agent Markup Language is defined using RDF/RDFS
- OIL is the Ontology Inference Layer
- W3C OWL is based on DAML+OIL
- Authoring tools are available



Tools to be used

The GSO Working Group will be based on OWL that has recently become a W3C Recommendation.

There are several tools available that support creation of DAML+OIL or OWL Ontologies:

OntoMat, Chimaera, PC Pack, Protégé and others

GSO WG will use Protégé from the University of Stanford as authoring tool.

Protégé supports OWL, is extensible, and based on HP Lab's Jena Package

Jena Package might be used to build services later too

Other tools for reasoning, inferring etc will be selected later.

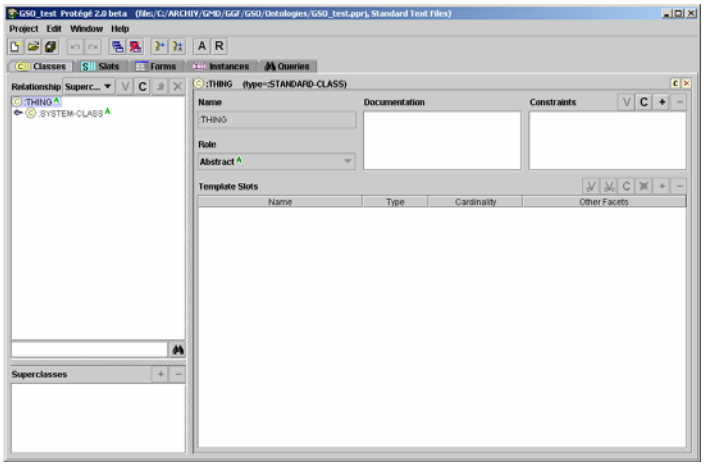
Protégé

Current Version with OWL plugin is 2.0 (stable beta)

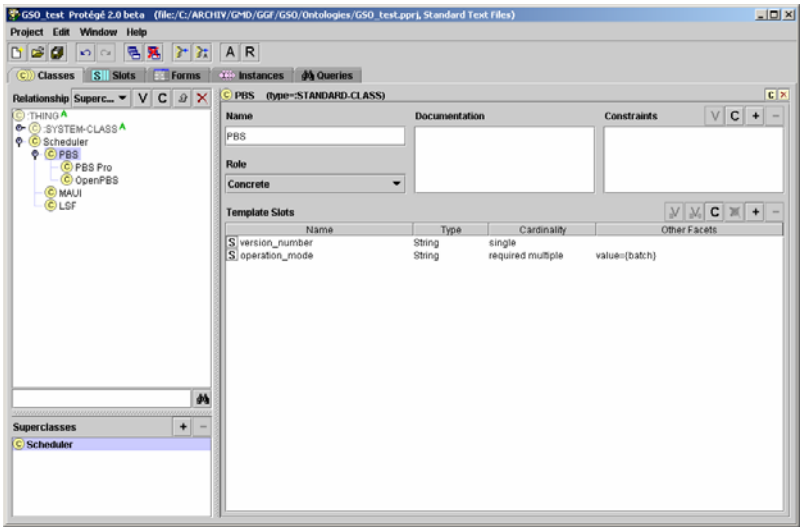
Public Domain Software with growing user group: more than 11k registered Users

more than 1600 discussion list members

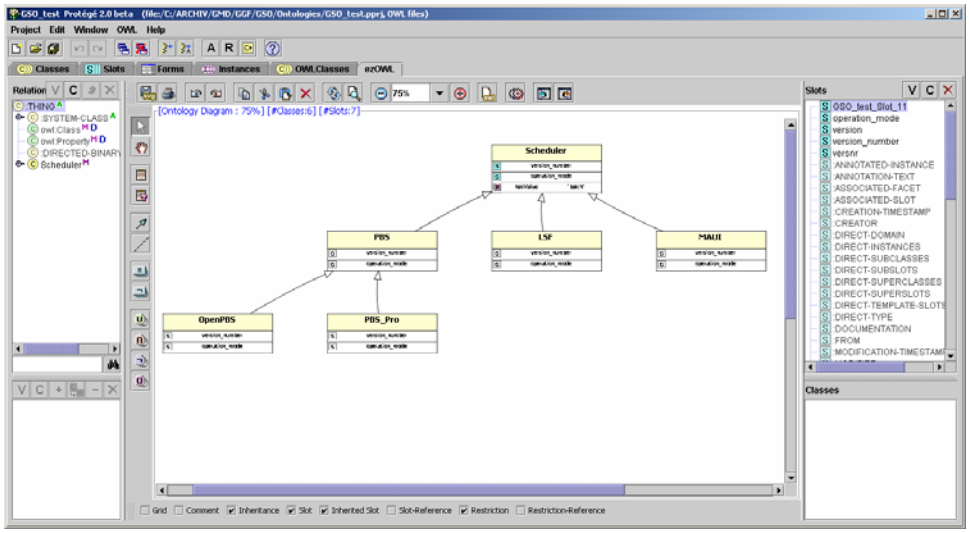
51 plug-ins



Protégé example



Protégé example - class diagram



OWL example

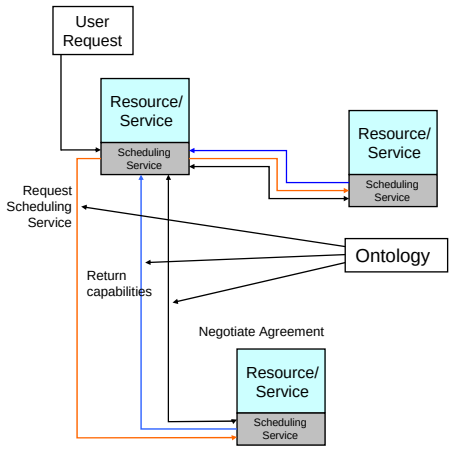
```
<rdf:RDF
    xmlns:rss="http://purl.org/rss/1.0/"
    xmlns:jms="http://jena.hpl.hp.com/2003/08/jms#"
    xmlns:j.0="http://protege.stanford.edu/plugins/owl/protege#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#"
    xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
  <owl:Ontology rdf:about="">
    <owl:imports rdf:resource="http://protege.stanford.edu/plugins/owl/protege"/>
  </owl:Ontology>
  <owl:Class rdf:ID="LSF">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Scheduler"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="OpenPBS">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#PBS"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="MAUI">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Scheduler"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Scheduler">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:about="#operation_mode"/>
        </owl:onProperty>
        <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >batch</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

```
  <owl:Class rdf:ID="PBS_Pro">
     <rdfs:label>PBS Pro</rdfs:label>
     <rdfs:subClassOf>
       <owl:Class rdf:about="#PBS"/>
     </rdfs:subClassOf>
   </owl:Class>
   <owl:Class rdf:ID="PBS">
     <rdfs:subClassOf rdf:resource="#Scheduler"/>
   </owl:Class>
   <owl:DatatypeProperty rdf:ID="GSO_test_Slot_11"
      rdf:type="http://www.w3.org/2002/07/owl#FunctionalProperty">
     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
   </owl:DatatypeProperty>
   <owl:DatatypeProperty rdf:ID="version_number"
      rdf:type="http://www.w3.org/2002/07/owl#FunctionalProperty">
     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
     <rdfs:domain rdf:resource="#Scheduler"/>
   </owl:DatatypeProperty>
   <owl:DatatypeProperty rdf:ID="versnr"
      rdf:type="http://www.w3.org/2002/07/owl#FunctionalProperty">
     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
   </owl:DatatypeProperty>
   <owl:DatatypeProperty rdf:ID="version"
      rdf:type="http://www.w3.org/2002/07/owl#FunctionalProperty">
     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
   </owl:DatatypeProperty>
   <owl:DatatypeProperty rdf:ID="operation_mode">
     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
     <rdfs:domain rdf:resource="#Scheduler"/>
   </owl:DatatypeProperty>
</rdf:RDF>
```

OWL output of the example from previous slides

Using the Grid Scheduling Ontology

- Finding Scheduling Services that are able to schedule the given task
- Matching properties/attributes/capabilities of Scheduling Services with user request
  - coping with different local names for attributes
  - different meanings of attributes, e.g. wall-clock minutes or cpu minutes
- Inference based matching
- Automatic classification of scheduling services



Charter of the proposed Grid Scheduling Ontology Working Group

Focus/Purpose:
This working group has the goal to produce an ontology of Grid Scheduling accompanied by a set of documents describing the ontology and the tools/libraries used to create the ontology and to make use of the ontology later.

Scope:
The ontology will be based on the Grid Scheduling Dictionary developed by the Grid Scheduling Dictionary Working Group earlier.

Goals:
The working group will create an ontology of the Grid Scheduling domain supporting the scheduling of Grid resources done by local and distributed instances of software subsystems like schedulers, brokers or corresponding Grid services. The ontology created will provide the machine processable meaning of scheduling terms and conditions that is needed to negotiate service level agreements between usually heterogeneous systems operated at different independent sites. The working group will define usage and hierarchy of terms from the Grid Scheduling Dictionary thus helping to understand these terms and enable tool builders to incorporate the ontology into their tools. The ontology will overcome the shortcomings of a dictionary allowing classification of schedulers, reasoning about schedulers or mapping semantics of different scheduling systems for example. Using the ontology generated by the working group when designing and implementing the next generation of Resource Management Systems and their corresponding Grid services may further lead to ontologydriven systems.

Charter of the proposed Grid Scheduling Ontology Working Group

Goals (continued):
The Grid Scheduling Ontology Working Group will establish a close collaboration with the Semantic Grid Research Group. While this research group addresses more general, long term research on the Semantic Grid, the Grid Scheduling Ontology Working Group will produce one concrete instantiation of an ontology. The Grid Scheduling Ontology Working Group will further collaborate with other working groups of the Scheduling and Resource Management Area, namely the Grid Resource Allocation Agreement Protocol Working Group and the Job Submission Description Language Working Group.

Class of documents: The intermediate documents will be of class Informational, the final one will be of class Community Practice.

Current Document: None.

Status: It is planned to hold a BoF at GGF-10.

Milestones

Milestones:

January 2004 :	Draft charter ready and posted to the GGF web site
GGF-10 WG meeting:	Discussion of the charter; Presentation of a draft ontology and the tools used; Planning of follow-up steps
GGF-12 WG meeting:	Integration of all terms selected from dictionary; Request for feedback
GGF-13 WG meeting:	Demonstration of usage of the ontology
GGF-14 WG meeting:	Grid Scheduling Ontology ready; Final document summarising and describing the ontology available; Description how the ontology may be used by resource management systems or services negotiating SLAs

Contacts to other Working Groups or Research Groups

Contacts already existing to

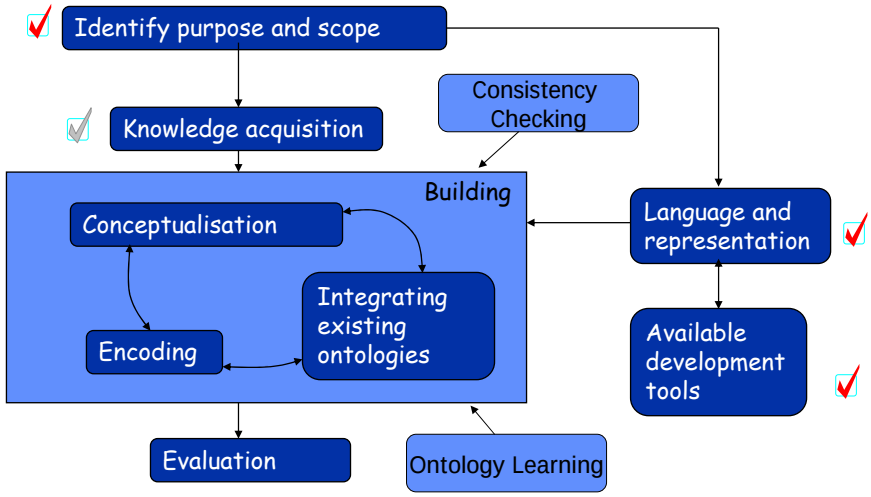
Semantic Grid Research Group
Grid Resource Allocation Agreement Protocol Working Group GRAAP

Groups we think contacts are useful

Scheduling Architecture Research Group
Job Submission Specification Language JSDL

others?

State of the work



Next Steps

Next WG meeting will be at GGF12 (Fall 2004)

GSO WG should receive approval within the next weeks.
We would then announce the WG within GGF,
and communicate Web-site and mailing-list

Work to be done prior to next meeting at GGF12:

- Prepare first draft of the structure of the Ontology
- Prepare an initial version of the ontology based on this draft and the Scheduling Dictionary
- Find out whether
  - ontologies already exist for this domain
  - they might be used when creating the Grid Scheduling Ontology

Contribute

How to start participating:

Join the GSO mailing list, attend the BOF
send mail to: majordomo@gridform.org with body: subscribe gso-wg

Commit yourself for activities of GSO working group

Volunteer to co-chair the WG

contact: Philipp Wieder ph.wieder@fz-juelich.de
Wolfgang Ziegler wolfgang.ziegler@scai.fraunhofer.de
http://www.gso-wg.org

Grid Scheduling Dictionary: http://forge.ggf.org/projects/sd-wg