Development of Front End Tools for Semantic Grid Services

Development and Integration of Semantic Component with Garuda Grid Portal

Dr. S. Thamarai Selvi,
Professor & Head,
Dept of Information Technology,
Madras Institute of Technology, Anna University, Chennai
Objective

- Developing Components for describing Garuda Grid Resources and their better discovery
- Integrating them with Garuda Grid Portal for identifying suitable resource for job submission

<table>
<thead>
<tr>
<th>Project Title</th>
<th>Development of Front End Tools for Semantic Grid Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Funding Agency</td>
<td>Centre for Development of Advanced Computing</td>
</tr>
<tr>
<td>Funded To</td>
<td>Madras Institute of Technology, Anna University, Chennai</td>
</tr>
<tr>
<td>Duration</td>
<td>2 years Since April 2005</td>
</tr>
<tr>
<td>Total Budget</td>
<td>12 Lakhs</td>
</tr>
</tbody>
</table>
Motivation

- Conventional mechanisms
  - UDDI
  - MDS
- They offer searching mechanism based on keywords.
- The node providers need to agree upon attribute names and values.
- In grid like environment, where resources come and go there is always a demand for framework to support semantic description and discovery of services and resources.
Background

Grid

A kind of distributed infrastructure that enables flexible, secure, coordinated resource sharing among dynamic collections of individuals, institutions, and organizational resources. (*This is what Virtual Organization is*)

Semantic Grid

The **Semantic Grid** is an extension of the current Grid in which information is given a well-defined meaning, better enabling computers and people to work in cooperation.
Related Tools

- **Ontology**
  
  You need an Editor to Create Ontology

- **Inference Engine**

  To retrieve Knowledge from Ontology
Ontology

- Ontologies are used to capture knowledge about some domain of interest.
- Ontology describes the concepts in the domain and also the relationships that hold between those concepts.
- Complex concepts can therefore be built up in definitions out of simpler concepts.
- **Web Ontology Language (OWL)** is widely used to create Ontology.
  
  Ex: Protégé, an OWL editor
Conventional Semantic Grid Service Architecture

Application Layer

Knowledge Layer

High Level Middleware layer

Core Middleware layer

Fabric Layer

Portlet

Discovery Portlet

Algernon reasoner

Query Generator

 Ontology Knowledge base

Matching Resource

Resource requester

 Portlet

Data Management

Information Management

Execution Management

Authorization

Authentication

Super Computer

Cluster

Desktop Machine

Storage Devices

Resources

Resources
Knowledge Layer

- Comprises two modules – **Semantic Description** and **Discovery**

**Semantic Description**

- Domain Knowledge of grid is represented in ontology template
- MDS is used to ‘plug’ grid resource information
- Protégé-OWL APIs are used to build knowledge base of the grid using ontology template

**Semantic Discovery**

- Algernon inference is used to retrieve resource information
Ontology Template

Definition – 1

Any resource can be modeled as an instance of a specific class provided that the resource can be described using the properties defined in that class.

Definition – 2

An ontology template is the domain specific ontology that provides hierarchy of classes with properties to define characteristics.

- Protégé-OWL APIs are used to describe grid resources in the ontology template.
Resource Ontology Template
Semantic Component

Description

- Grid Resources
- Ontology Knowledge Base
- Resource Advertisement
- MDS

Discovery

- Matchmaking Algorithm
- Resource Request Specification
- User
Sequence of Operations

Description

- This module aggregates available grid resource information through respective middleware component and creates knowledge base using the pre-defined ontology template.

Discovery

- This module implements a matchmaking algorithm that uses algernon inference engine to interact with the knowledge base.
Semantic Description

- GIIS service runs on globus machine will retrieve resource information of the local host and stores it in LDAP server from where we can query the information.

- Protégé-OWL provides versatile libraries with which one can manage ontology and knowledge base. With those APIs insertion and removal of resources are possible.

```java
OWLNamedClass computerC=owlModel.getOWLNamedClass("WorkStation");
OWLDatatypeProperty hasIP = owlModel.getOWLDatatypeProperty("hasIP");
cpuI.addPropertyValue(owlModel.getOWLObjectProperty("hasCPUVendor"),cVendorI);
computerI.addPropertyValue(owlModel.getOWLObjectProperty("hasCPU"),cpuI);
```
The Query format for a resource request whose Requirement criteria is RAM:500 and OS:Linux is
“freeRAM: > 500 hasOS: Linux”

Corresponding Algernon Axiom

((instance RAM ?inst)(hasFreeMB ?inst ?val)
(hasOS ?inst “Linux”)(TEST(:LISP(=?val"+rightTag+")))
presentInComputer?inst ?instanceComputer)).

If the user request Unix OS and if it is not available, the discovery Mechanism obtains machines with Linux OS as the Linux concept is modeled as subconcept of Unix.
Aggregation of heterogeneous and geographically distributed resources such as computing, storage and special equipments from various research labs and Academic institutions across the country
Our Contribution to Garuda

- The Ontology template has been modified to suit the Garuda infrastructure.
- The discovery mechanism is extended to support the discovery of Garuda resources.
- The resources are semantically described using the protégé-OWL libraries from the registry containing resource information maintained in Garuda.
Constraints with Garuda

- A plain text file with specific format contains the resource information.
- Garuda is capable of scheduling jobs to Globus as well as Moab metascheduler.

Hence

- We developed a resource description module that reads the text file and builds knowledge base onto the ontology template.
- A discovery module that accepts input from Garuda job submission portal and discovers suitable resource by searching into the knowledge base.
Semantic Component Integrated with Garuda
‘Garuda’ uses semantic grid technology for its resource discovery.

The ‘Garuda’ resource discovery portal can understand the ‘meaning’ of the type of resource requested and retrieves closely matching resources, if exact resource is not available.

For Ex, if the user request a resource with AIX OS and Release 5.0, the discovery portal retrieves AIX machines with 5.1 release if the machine with 5.0 release is not available. This is because, the portal understands that release 5.1 has backward compatibility and hence it can run the jobs compiled with 5.0. (The concept of ontology establish this relationship that can be reasoned using algernon inference engine).
Resource Request Form

Select Operating System: Linux
Select Release: 2.2
Processor Type: Intel
Select Search Level: Compatible Release

Submit

List of Matching Resource(s)
- MIT/Linux_2.6
- Bangalore/Linux_2.4
- Pune/Linux_2.4

Select
Reset
Current Status

- Implementation of semantic components has been completed
- Integration and Testing the Semantic Component with Garuda Grid Portal was completed
- Preparation of Documents related to projects covering User Manual, Design documents, Test cases are underway. First draft has been submitted and corrections are carried out

- Man Power Trained - 3
- Publications - 5 (International Conferences)
- Journal -1 (Submitted under review)
Publications


References


Questions