

An Ontology-Based Host Resources Monitoring Approach in Grid Environment*

Yijiao Yu and Hai Jin

Cluster and Grid Computing Lab,
Huazhong University of Science and Technology, Wuhan, 430074, China
yjyu@mail.ccnu.edu.cn
hjin@hust.edu.cn

Abstract. Monitoring the real-time status of voluntary nodes is a basic task of *Quality of Services* (QoS) management in grid. The heterogeneity of distributed host resources is an obvious obstacle of grid resources monitoring. An ontology-based approach is presented in this paper to help monitor host resources, focusing on integrating and sharing the status information of grid resources. The ontology designed is originated from *Management Information Base* (MIB) in network management. The modeling methods are presented in detail, including data type translation rules, class and properties translation rules, and an ontology is proposed in this paper.

1 Introduction

As a large-scale distributed system, grid discovers unemployed resources in network, collects idle resources, and provides computing and storage services to users. If the real-time status of employed resource of all voluntary nodes can be represented in easily readable and understandable format to the load balancing subsystem of grid, the scheduler is able to assign the requests from users to the idle and appropriate nodes, and the QoS of grid is improved [1]. Due to the heterogeneity of host resources, the job scheduler of grid has difficulty in understanding and integrating the status reports from each node in various syntax formats and semantic representation.

Ontology is a powerful knowledge representations and management tool, and it is popular in the heterogeneous information management in web service, semantic web and grid applications [2]. Some grid monitoring systems have been built [3][4], however, the uniform and sharable representations of host resources are not discussed. The semantic representations, with ontology, of host resources are presented, and some encouraging results have been reported in [5]. In this paper, an ontology-based host resources monitoring approach in grid environment is proposed.

The remainder of this paper is organized as follows. Section 2 shows the modeling principles of the host resources ontology. Section 3 illustrates the translation rules from SMI to OWL. Section 4 presents the ontology. Section 5 concludes this paper with additional comments.

* This work is supported by National Basic 973 Research Program of China under grant No.2003CB317003.

2 Build the Ontology from MIBs

Ontology is an explicit specification of a conceptualization where definitions associate concepts, taxonomies, and relationships with human-readable text and formal, machine-readable axiom. It is not difficult to define some concepts and the relationships, but how to define a minimal and complete set of concepts and relationships is a challenge. The principles of our ontology modeling are listed below.

First, the concepts and properties in ontology of host resources should originate from some existing open network devices monitoring and management standards or protocols. The completeness of the concepts, properties and constraints has been discussed for a long time during the standards proposing and improving period. In addition, the concepts and properties in ontology, extracted from the existing and popular protocols, are easy to be accepted and shared by others. Due to the popularity of *Simple Network Management Protocol* (SNMP), most of the nodes in grid support SNMP agents. Up to now, more than 100 *Request for Comments* (RFC) documents, related to MIBs, have been proposed [6] and at least 15 RFC documents, with the “standard” or “proposed standard” status, are related to host resources. Therefore, it is necessary to investigate the MIBs of host resources before modeling.

Second, ontology should describe the host resources in grid completely, and the number of concepts should be as few as possible. All the applications, winners of the Semantic Web Challenge 2003, use simple ontologies, and few ontologies contain more than 100 concepts [7]. According to this experience, we check the MIBs and find that some concepts are not valuable in grid environment. Although they are useful in the traditional network management scenario, these concepts or properties are discarded during our ontology modeling.

3 Converting SMI MIBs to OWL

Although MIBs are expected to be translated into lightweight ontology only with the classes and properties in [5], the detailed methods of translating MIBs into RDF is not illustrated. The translation rules from *Structure of Management Information* (SMI) MIBs to OWL are categorized into data type, class and semantics constraints.

3.1 The Translation Rules of Data Types

Mapping data types from SMI to OWL is the first step of the host resources ontology modeling. The data structure in MIBs must be one of the following: a base type, the BITS construct, or a textual convention [8]. Furthermore, conceptual tables and conceptual rows are frequently utilized to describe complex resources. The translation rules about the four kinds of data types are illustrated.

The data types of leaf nodes in MIB tree belong to “base types”, and all kinds of data types in “base types” are listed in the first column of Table 1. There is no specific definition of data types in the latest OWL specification, and the RDF mechanism for data types is inherited by OWL [9]. OWL uses most of the built-in XML schema data types and more than 35 basic data types are recommended for use with OWL, such as `xsd:string`, `xsd:integer`, `xsd:unsignedLong`.

Table 1. The mapping rules of base data types in SMI MIBs

Data types in SMI	Domain	Data types in OWL
INTEGER	-2147483648..2147483647	xsd:integer
OCTET STRING	String (SIZE (0..65535))	xsd:string
OBJECT IDENTIFIER	0..4294967295	xsd:nonNegativeInteger
Integer32	-2147483648..2147483647	xsd:integer
IpAddress	String (SIZE (4))	xsd:string
Counter32	0..4294967295	xsd:nonNegativeInteger
Gauge32	0..4294967295	xsd:nonNegativeInteger
TimeTicks	0..4294967295	xsd:nonNegativeInteger
Unsigned32	0..4294967295	xsd:nonNegativeInteger
Opaque	OCTET STRING	xsd:string
Counter64	0..18446744073709551615	xsd:unsignedLong

If a domain of data type in SMI is identical to a domain in OWL, the two data types can be mapped directly. For example, “INTEGER” in SMI is from -2147483648 to 2147483647, while “xsd:integer” in OWL is also within the same domain. Therefore, “INTEGER” in SMI is translated into “xsd:integer” directly. Some data types in SMI can not find the exactly identical data types in OWL directly, e.g. “IpAddress”. To simplify the translation rules, these data types in OWL is selected to be the target data types of translation, which is the minimal closure of the data types in SMI. As an example, “IpAddress” is represented with string in SMI, no more than 4 characters, and mapped to “xsd:string”. This translation enlarges the representation domain of the data types in SMI. However, with specific restrictions of the domain when developing ontology, we can make them equally. Under the two principles, all the target data types of “base types” are listed in the third column in Table 1.

The BITS construct represents an enumeration of named bits. This collection is assigned non-negative and contiguous values, starting at zero. Only those named-bits enumerated may be present in a value [8]. As the translation method of “IpAddress”, the BITS construct can be mapped to xsd:nonNegativeInteger with some limitations.

3.2 Textual Convention

A textual convention is a newly-defined type defined as a sub-type of a base type. With comparison to “base types”, each of these textual convention data types has a different name, a similar syntax, but a more precise semantics. All the textual convention data types can be represented in “base types”, but the domains are limited. There are two possible ways to represent the textual conversation data types in OWL.

The first one is only to map these textual convention data types to basic data types in OWL, so necessary restrictions of domain should be attached. For example, “DisplayString” is a textual data type in SMI, which represents “OCTET STRING (SIZE (0..255))”. It is mapped into “xsd:string” with the special limitations of minimum and maximum lengths.

```

<xsd:simpleType name="DisplayString">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="255" />
  </xsd:restriction>
</xsd:simpleType>

```

The second way is to define some new data types in OWL. Users can define new data types in OWL if necessary. For example, we define a “DisplayString” data type, whose maximum length is no more than 255, with XML schema above. In this paper, the first way is applied because the direct mapping, using the public data types, is helpful to share the application ontology with others.

3.3 Conceptual Tables, Conceptual Rows, Class and Properties

Tabular structures on an ordered collection of objects, conceptual table and row, can be regarded as complicated object types. Each conceptual table contains zero or more rows, and each row may contain one or more scalar objects, termed columnar objects [8]. An example of conceptual table, “hrStorageTable”, is illustrated below. “hrStorageEntry” is the conceptual row of this table, which depicts of a logical storage of hard disk, including the “hrStorageSize”, “hrStorageUsed”.

```

hrStorageTable OBJECT-TYPE
  SYNTAX      SEQUENCE OF HrStorageEntry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    .....
 ::= { hrStorage 3 }

```

Although conceptual tables and conceptual rows are data types in SMI, they are not translated into basic data types in OWL, because the identifiers of the conceptual tables and conceptual rows are internal nodes in MIB tree. In general, the object identifier, whose data type is conceptual row, is translated into class, and multiple rows are multiple instances of the specific class. In ontology, zero or multiple instances of conceptual rows are contained in the class of conceptual table.

Classes and their properties are the basic elements of ontology. In MIB tree, the child nodes describe the parent nodes from different aspects. Usually, internal nodes should be translated into classes in ontology, but not all internal nodes should be translated into classes in the modeling. The managed objects in MIBs are grouped and organized from the view of network management and they are not completely suitable for grid resources monitoring. However, we concentrate on the dynamic device utilization status in grid monitoring, so they are reorganized in host resource ontology modeling, which will be illustrated in Section 4. In this way, the number of classes decreases dramatically during ontology modeling.

4 The Ontology

Describing the relationships between classes in ontology is difficult and necessary in ontology modeling. There are several kinds of relationships supported by OWL, such

as subclass, equivalence. To simplify the relationships of classes in host resource ontology of grid, we discard the complete path of node in MIB tree, only integrating the root nodes in the RFC documents related host resources. The relationships of classes in host resources ontology are shown in Fig. 1.

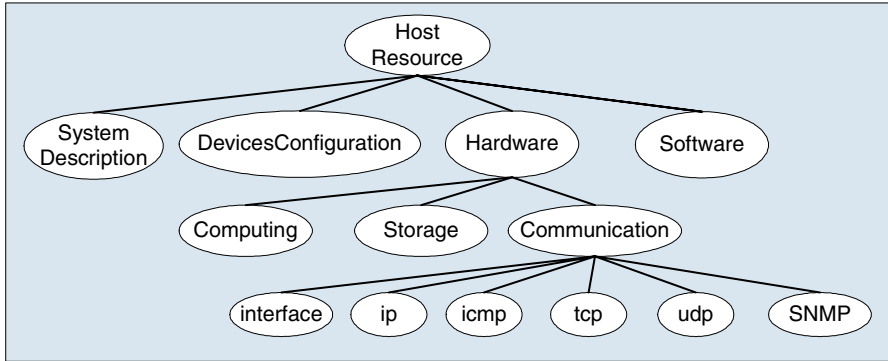


Fig. 1. Part of the host resources ontology

“HostResource” is a basic concept in the host resources ontology, which contains “SystemDescription”, “DevicesConfiguration”, “Hardware”, and “Software”. In host resource monitoring applications, the real-time status description of a voluntary node is an instance of this class. Therefore, if there are n computers in grid, there are n instances of “HostResource”. The class of “SystemDescription” gives the basic descriptions of a host, which is an aggregation of “hrSystem” subgroup in RFC 2790 and “system” subgroup in RFC 1213. All the slots of “SystemDescription” are leaf nodes in MIB tree with simple data types, e.g. “hrSystemUptime”, and “sysName”.

“DevicesConfiguration” is the static description of the hardware configuration in a host, such as the types of hard disk, printer, CPUs and video cards. The hardware configuration information is organized as child node of “hrDevice” in RFC 2790, but it is extracted as a separated class contained in “HostResources” directly. In fact, the relationships between classes in Fig. 1 are not the same as that of MIB tree. The hardware resources are categorized into computing resources, communication resources and storage resources. The values of the slots of these classes are dynamically changing when the grid nodes runs. For example, the traffic of network interfaces increases continuously and the utilization of CPUs is varying. These dynamic statuses should be monitored periodically. The MIBs of WWW, E-mail, and DBMS can be translated into classes as the MIBs of hardware resources. Moreover, the specific MIBs of grid can be defined and integrated into the ontology to provide the powerful description ability of host resources.

5 Conclusion

In this paper, we propose an ontology-based host resources monitoring approach in grid environment and describe the ontology with OWL. With the ontology, grid moni-

tors and schedulers can obtain better description, data interoperability and integration ability of the physical resources in the grid. Moreover, having host resources semantically annotated, enables us to perform semantic matching which significantly improves query results and delivers a ranked list of best matching candidates for a given service requirement from grid users. In general, the ontology-based host resources monitoring method brings a new approach to represent semantics of the host resources in grid and gives hints for the future research on semantic grid QoS.

References

1. D. A. Menasce and E. Casalicchio, "QoS in Grid Computing", *IEEE Internet Computing*, 4 (2004):85-87.
2. M. Cannataro and D. Talia, "Semantics and Knowledge Grids: Building the Next-Generation Grid", *IEEE Intelligent Systems*, 1 (2004):56-63.
3. W. Smith, "A System for Monitoring and Management of Computational Grids", *Proceedings of 2002 International Conference on Parallel Processing (ICPP'02)*, (2002):55-64.
4. B. Tierney, B. Crowley, D. Gunter, M. Holding, J. Lee, and M. Thompson, "A Monitoring Sensor Management System for Grid Environments", *Proceedings of the 9th IEEE Symposium on High Performance Distributed Computing*, (2000):97-104.
5. J. E. López de Vergara, J. E. L. Villagra, and V. A. Berrocal, "Applying the Web ontology language to management information definitions", *IEEE Communications Magazine*, 7(2004):68-74.
6. RFC INDEX, http://www.ietf.org/iesg/1rfc_index.txt.
7. M. Klein and U. Visser, "Semantic Web Challenge 2003", *IEEE Intelligent Systems*, 3 (2004):31-33.
8. K. McCloghrie, D. Perkins, and J. Schoenwaelder, RFC 2578: Structure of Management Information Version 2 (SMIv2), (1999), <http://www.ietf.org>.
9. D. L. McGuinness and F. van Harmelen, "OWL Web Ontology Language Overview", (2004), <http://www.w3.org/TR/2004/REC-owl-features-20040210>.