

Design and Implementation of Semantic Resource Management Platform based on LDAP

K.Q. Wu, C.C. Liu

Institute of Network Technology, BUPT,
Beijing

Abstract—With the development of information technology, nearly all resources are presented in an informational way. But it's uncommon to manage resources platforms with resource semantic annotation. LDAP (Lightweight directory access protocol) is applied widely for its simplicity and integrity. OWL (Ontology Web Language) is widely used in the definition of ontology along with the continually maturation of semantic web. REST (Representational state transfer) is a good solution to use when developing business service due to its lightweight property. Based on LDAP and ontology modeling, this paper provides an approach to design and implement a Semantic Resource Management Platform (SRMP), and opens the CRUD (create, retrieve, update, delete) operation of resources with restful style.

Keyword: LDAP, SRMP, CRUD

I. INTRODUCTION

Along with the development of information technology and the convergence of three networks, the need of knowledge reuse and sharing is getting more and more urgent. Thus developing reusable ontologies is of great significance to encourage the sharing and reuse of knowledge.

In the age of information explosion, an organized platform is urgently needed to manage information in various fields, directory service emerge as the times require. With this kind of service, we enable users to get the information needed, based on certain index mechanism.

To make the network resources understandable by machine, a resource-management website has been built in this paper, by combining LDAP and ontologies. Meanwhile, some restful resource-CRUD services are opened to different kinds of clients according to their authority.

This paper is organized as follows. First, in Section 2, the key technologies used in this paper is introduced. In Section 3 and 4, the design and implementation of SRMP is presented. In Section 5, summarization and conclusion are given towards the whole paper.

II. KEY TECHNOLOGY ANALYSIS

A. LDAP

1) Brief introduction to LDAP

LDAP is a lightweight directory access protocol, which is based on X.500 standard, but simpler and can be customized to users' need. Moreover, LDAP supports TCP/IP, which is necessary for visiting Internet.

LDAP is used to provide information services which are known as directory services. It stores data with hierarchical formats. The root is called "Base DN", such as "dc=mydomain,dc=subdomain" or "o=mydomain.subdomain". The preceding method is usually used, which can be found in Windows AD, Oracle Directory Server and so on.

DN (Distinguished Name) is used to distinguish one record from others in LDAP. The part which is in the position "leaf" is named RDN (Relative Distinguished Name), such as "wkq" in the record "dn: cn=wkq,ou=bupt,dc=china,dc=world", which must be equal.

LDAP and database are quite similar, but with fundamental differences. A database is designed to make reading and writing convenient, while LDAP optimizes the reading specially, which is similar to phonebook. Thus, it can't be used to store data which would be inserted frequently. Meanwhile LDAP is just a protocol, which doesn't involve how to store these information. So a back-end database component is needed to realize the storage, which can be BDB (Berkeley DB), ldbm, shell, passwd and so on.

2) Advantages of LDAP

- Cross Platform: LDAP is an Internet standard protocol, which has been widely accepted by industry. It can be seen as a passport to access all authenticated directory-services while ignoring the provider of the services.
- Economical: no need to pay for every client connection or permission.
- Efficiency of Query: reading optimization.

B. OWL

OWL was published by W3C in July 31th, 2002, which has three sub languages as follows:

- OWL Lite: works for clients who only need one classification structure and simple constrains among attributes.
- OWL DL: supports for clients who need the representation on reasoning-system in the maximal degree.
- OWL Full: suitable for people who wants to express on RDF with no computation guaranteed in the maximal degree. No reasoning software can support all the features of OWL Full.

In semantic webs, everyone may want to create or modify ontologies, so readability and easily use are very important for an ontology language. Instead of describing logical, OWL combines every class's information according to framework organization. This makes ontology easier to un-

derstand. Protégé is exactly the kind of ontology-design tool, besides, it's "open source", which can be very easy to extend. Therefore, SRMP chooses it to model the resources.

C. REST

1) Brief introduction to REST

REST (Representational State Transfer) is a software architecture style which was proposed by Doc. Roy Fielding in his paper in 2000. At present, more and more web-services developments are designed and realized with restful style since that REST is obvious more concise than SOAP and XML-RPC.

Four characters about REST:

- Addressability: an addressable application may provide each information an URI since that resource is exposed through URI.
- Stateless: each HTTP request is isolated.
- Connectivity: resources can be connected with their representations.
- Unified Interface: bases on HTTP protocol, unified as four common operations, GET, POST, PUT and DELETE.

2) Advantages of REST

- Unified interface & concise architecture.
- Can increase response speed using cache.
- High expansibility
- Only bases on HTTP protocol, lightweight.
- May simulate client request with browser.

III. DESIGN OF SRMP

A. Structure design

Resources are divided into two categories:

- Sensor: has perception ability, such as temperature sensor, humidity sensor, and camera and so on.
- Entity: has no perception ability, but can be monitored or sensed.

As we all know, the resources of the world may have "superior-subordinate" relations, "inclusion" relations, or "similar" relations. Similarly, the relation between resources can be mapped onto the sensor and entity. Therefore, the conception of "sensor relationship", "entity relationship" and "sensor-entity relationship" can be provided.

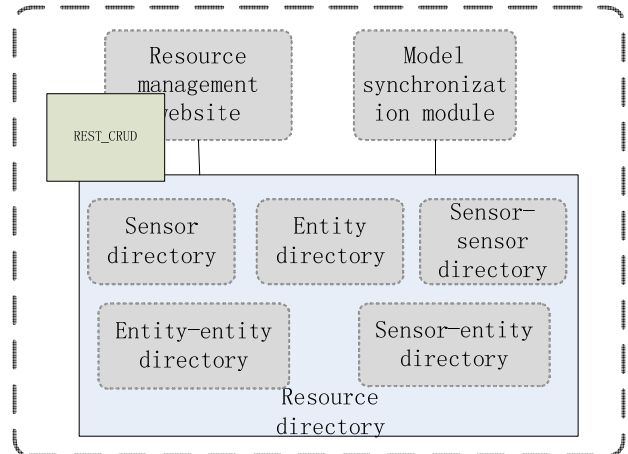


Figure 1 SRMP STRUCTURE

As show in Figure 1, there are three modules in SRMP:

- Resource directory: LDAP storage. The whole resource directories are composed by five sub directories, which are sensor directories, entity directories, sensor-sensor directories, entity-entity directories, sensor-entity directories.
- Resource Management Website: realize with Spring MVC technology.
- Model Synchronization Module: mapping onto directory modules after semantic modeling.
- REST_CRUD: that is, packing the creating, retrieve, updating and deleting of resources as RESTful web services, then open them to different kinds of clients with different authority.

B. Semantic Resource Modeling

The so-called semantic resource modeling, that is constructing ontologies needed. Specific construct steps as follows:

- Consider reusing existing ontology, such as time ontology.
- List important terms of the ontology, in this case, which can be "sensor", "entity", etc.
- Define the "class" and "class integration", "attributes" and "relationships", "attributes restriction".

TABLE 1 RESOURCE MODELING.

```

<?xml version="1.0"?>
<!DOCTYPE Ontology [
  <!ENTITY xsd "http://.../XMLSchema#" >
  <!ENTITY xml "http://.../namespace" >
  <!ENTITY rdfs "http://.../rdf-schema#" >
  <!ENTITY rdf "http://.../rdf-syntax-ns#" >
]>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.semanticweb.org/wukeqiao/
  ontologies/2014/9/Resource"
  .....
wukeqiao/ontologies/2014/9/Info">
  <Prefix name="rdf" IRI
    ="http://www.w3.org/1999/02/22
    -rdf-syntax-ns#" />
  <Prefix name="rdfs" IRI
    ="http://www.w3.org/2000/01/rdf-schema#" />
  <Prefix name="xsd" IRI
    ="http://www.w3.org/2001/XMLSchema#" />
  <Prefix name="owl" IRI
    ="http://www.w3.org/2002/07/owl#" />
  <Declaration>
    <Class IRI="#Entity" />
  </Declaration>
  <Declaration>
    <Class IRI="#Guard" />
  </Declaration>
  <Declaration>
    <Class IRI="#Sensor" />
  </Declaration>
  <Declaration>
    <Class IRI="#Thermometer" />
  </Declaration>
  <Declaration>
    <ObjectProperty IRI="#id" />
  </Declaration>
  .....
  <ObjectPropertyRange>
    <ObjectProperty IRI="#zhName" />
    <DataAllValuesFrom>
      <DataProperty abbreviatedIRI
        ="owl:topDataProperty" />
      <Datatype abbreviatedIRI="xsd:string" />
    </DataAllValuesFrom>
  </ObjectPropertyRange>
  .....

```

For brevity, only part of the semantic modeling list in Table 1.

C. Resource directory design

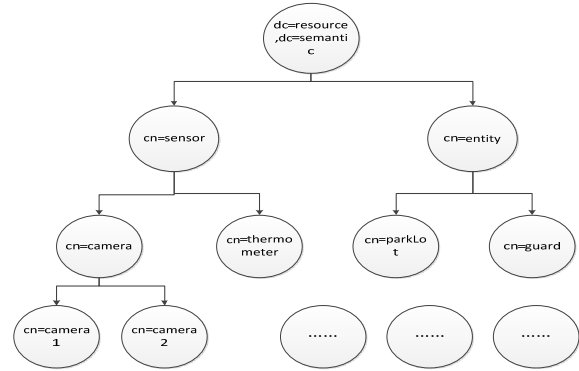


Figure 2 LDAP resource directory model

As show in Figure 2, “dc=resource,dc=semantic” is taken as the top-level directory, that is the top suffix. Sensor-sensor, entity-entity and sensor-entity relationships can be represented through the sense attributes and relation attributes of sensor and entity. So there is no need to create DN likes “cn=sensor_sensor” or something like that. More details can be seen in Section 4.

IV. SRMP REALIZATION

A. Resource directory deploy

The current mainstream LDAP servers are as follows:

- OpenLDAP: open source, can be compiled to run on multiple platforms.
- Novel Directory Server: can run on Netware, Windows NT/2000, Solaris, Linux, etc.
- IBM Directory: be able to run on AIX, HP-UX, Linux, Solaris and Windows 2000.
- Microsoft AD (Active Directory) and ADAM (Active Directory Application Model): both are directory service products of Microsoft, which could run under Windows XP Professional and Windows 2003 of course. ADAM can be seen as a simplified version of AD. But AD is mainly about network infrastructure and needs NDS, while ADAM often works for applications and doesn't need NDS.
- Oracle Directory Server: first known as Netscape Directory Server, then known as Sun ODE Directory Server, iPlanet Directory Server. Lately, renamed with Sun Java System Directory Server. Finally, purchased by Oracle. It can run on Windows, Solaris, Linux, etc.

Considering that ODS (Oracle Directory Server) allows object class modification in LDAP, which is needed to realize semantic. ODS is the best choice of deploying our directory services. Plus, it works well in both Windows and Linux system.

After the deployment of directory services, we analyze the ontologies modeled by protégé with Jena framework, and map them onto the server.

B. Semantic modeling map onto resource directory

Spring-REST architecture is widely used for developing restful web services, which have two meanings in this paper:

- Semantic modeling map onto resource directory.
- Resource’s CRUD operations, provided for authenticated third party.

TABLE 2 SRMP REST INTERFACES

Method	Interface	Parameters
Model map	serverRoot/templateMap	TemplateInfo
Instance map	serverRoot/instanceMap	InstanceInfo
Instance modify	server-Root/modifyInstance	TemplateInfo
Resource remove	serverRoot/deleted	URI (String)
Resource search	serverRoot/search	Text (String)
.....		

See in Table 2, serverRoot is used to represent http://serverip:8080/InforDir/, while serverip stands for the ip address that the server deployed services. TemplateInfo and InstanceInfo are defined as in Figure 3.

TemplateInfo	InstanceInfo
+zhName : string	+zhName : string
+type : string	+type : string
+classID : string	+classID : string
+pID : string	+pID : string
+uri : string	+uri : string
+pURI : string	+pURI : string
+baseProperty	+properties
+senseProperty	

Figure 3 TemplateInfo & InstanceInfo

C. Automatic deployment of directory structure.

To realize one-step deployment, five required directories (sensor directory, entity directory, sensor-sensor directory, entity-entity directory, sensor-entity directory) structures are coded. First, we create all required “semantic attributes” (see in Table 3), then build “object classes” for the directories (see in Table 4), and then establish the entry information (see in Table 5).

TABLE 3 ATTRIBUTES CREATION

```
ldapConnection = connectLDAP(usr, password, ip, port);
ldapConnection.authenticate(ROOT_DN, ROOT_PASSWORD);
String[] attributes = {"id", "baseProperty", "relationProperty", "senseProperty", "resourceId", "entityId"};
for attr in attributes
newAttrType = new LDAPAttributeSchema(attr, attr+"-oid", "must", cis, false);
newAttrType.add(ldapConnection);
dirSchema.fetchSchema(ldapConnection);
```

TABLE 4 OBJECT CLASS CREATION

```
LDAPObjectClassSchema sensor = new LDAPObjectClassSchema (
    "sensor",
    "sensor-oid",
    "new String[] {“top”}", // super class
    "sensor description", //des.
    "new String[] {“cn”, “id”, “uri”}", //must attrs.
    null, // optional attrs.
    false, //not obsolete
);
```

TABLE 5 ENTRY BUILD

```
LDAPAttributeSet attributeSet = new LDAPAttributeSet();
attributeSet.add(new LDAPAttribute("objectclass", "sensor"));
attributeSet.add(new LDAPAttribute("cn", "sensor"));
attributeSet.add(new LDAPAttribute("id", "1"));
String suffix = "cn=resource," + BASE_CN;
```

D. Resource management website



Figure 4 Resource Management Website

According to Resource Management Website (see in Figure 4), authenticated users can browse all the resources and details of them. Besides, they could do some operations such as creating, retrieve, updating and deleting based on their authority.

V. CONCLUSION

Based on analyzing LDAP, OWL and restful web service technologies, a semantic resource management platform based on LDAP has been designed and implemented.

SRMP successfully solved the problem of heterogeneous data through mapping semantic modeling onto resource directory. Besides, the representation of resources managed by SRMP is more effective with the organic combination of ontology and LDAP. In the end, the resource management website provides a friendly way for users to manage resources. In order to implement knowledge reuse and sharing, the CRUD operations of resources also have been encapsulated as restful web services, which can be visited by all authorized users.

In the future, more resources can be invoked on SRMP, in the same time, the stability of the management of different kinds of resources can be enhanced.

VI. ACKNOWLEDGEMENT

This research is supported by the National Grand Fundamental Research 973 Program of China under Grant No.

2011CB302506, 2012CB315802; National Key Technology Research and Development Program of China (Grant No. 2012BAH94F02); National High-tech R&D Program of China (863 Program) under Grant No. 2013AA102301; National Natural Science Foundation of China under Grant No.61132001,61372115); Program for New Century Excellent Talents in University (Grant No. NCET-11-0592); Project of New Generation Broad band Wireless Network under Grant No. 2014ZX03006003,2011ZX03002-002-01, 2012ZX03005008-001; The technology development and experiment of innovative network architecture(CNGI-12-03-007).

REFERENCES

- [1] Gerald Carter. LDAP System Administration [M]. USA: O'Reilly, 2003. 12-85.
- [2] M Wahl, T Howes, S Kille. Lightweight Directory Protocol (v3)[S], RFC 2251, 1997.
- [3] Richardson, L.& R.Sam.RESTful Web Services.2008
- [4] OWL-S1.1Release.<http://www.daml.org/services/owl-s/1.1/overview>.2004
- [5] OWL-S1.2Release.<http://www.ai.sri.com/daml/services/owl-s/1.2/>.2008