Multi-layer Grid Portal Architecture Based on Mobile Agent

Fang Juan
College of Computer Science, Beijing University of Technology,
100022 Beijing, China
fangjuan@bjut.edu.cn

Abstract

Many Grid portal architectures have three layers: browser, portal and Grid environment. We design a multi-layer Grid portal architecture, which introduced Mobile Agent; the mobile agents can gain the different information of the Grid resources. This paper analyzes the multi-layer architecture of Grid portal, presents the security strategy and introduces the function of different mobile agent.

1. Introduction

Grid portals are a class of www application servers that provide a secure online environment for gathering information about Grid services and resources as well as provide tools for utilizing these Grid services and resources to perform useful tasks [1].

Generally, Grid portal has three-tier. The users access Grid portal by browser, and then deal with the user’s request according to web request; it will be transferred to Grid I/O operation. The bottom layer is Grid environment, including Grid resources, Grid software and so on. To obtain the information of all the heterogeneous resources, we added a layer between the Grid resources and Portal layer, it is Mobile Agent layer. The mobile agents can gather all the information of the Grid resources, and the introduction of mobile agent technology reduces the network traffic during the grid resource discovery and monitoring process effectively.

2. Architecture

Our proposed architecture consists of various layers. Fig. 1 illustrates a five-layer architecture for the system.

**Client Layer:** The top layer is client layer, it is the end user who interacts with the system through a Web browser. The client layer consists of the end-user’s workstation running a web browser.

**Portal Layer:** In portal layer we use GPDK as the tool to develop the primary function. GPDK is a toolkit for developing grid portal; it provides a series of core components to access grid services by JavaBeans [2]. GPDK Java Beans are grouped into five categories as follows: Security, User profiles, File Transfer, Job Submission and Information Services.

**Security Layer:** In the field of information security, three core principles are confidentiality, integrity and availability. Although there are many facets to portal security, in our architecture, we primary considers the security features of the different portal

**Resource Layer**

- Storage Resources
- Information Resources
- Computational Resources

**Mobile Agent Layer**

- Sensor Agent
- Register Agent
- Update Agent
- Information

**Figure 1 – Multi-layer Grid portal architecture**
implementations in four key areas: authentication, authorization, auditing and session management [3].

**Mobile Agent Layer:** This layer is a special layer in the architecture. Actually, the mobile agents here are responsible for achieving the status of different dynamic resources and different Grid nodes.

**Resources Layer:** The resources in the grid environment are geographically distributed and each of them is owned by a different organization. Each of them has its own resource management mechanisms and policies and may charge different prices for different users necessitating the need for the support of computational economy in resource management [4].

3. Related Mobile Agents

Mobile agents provide an important paradigm for supporting dynamic services in Computational Grids. A mobile agent is a software module able to migrate among the hosts of a network, in order to carry on a specific task [4]. In this architecture, we use IBM’s Aglet software package as runtime environment of mobile agent, in the run process it can create many static and mobile agent instances, and then it can achieve the nodes’ status of grid environment.

We have two ways to define the agents. The first way is to divide the agents into two kinds. One is static agent, the other is mobile agent. The second way is to divide the agents by the function of the agents, such as information agent, computational agent and semantic agent. The primary agents and their function of the system are listed as follows.

**Sensor agent:** Monitoring the status information of local resource. The transmission of message object is implemented by massage transmission manner. It doesn’t run the collect information command, but return the result by calling the methods of basic class. Sensor agent acts as static agent to stay at agent runtime environment, waiting for the Update agent to call it. Sensor agent includes many agents, and achieves different resource status information respectively.

**Update agent:** Calling Sensor agent to update the status of nodes, and then achieve all the real-time data and status information. It also acts as static agent to stay at agent runtime environment to run update in turn.

**Register agent:** Grid nodes may use it to register, and unregister, if the static information of node changed, it can be updated by Register agent. The local node network information can be achieved by Sensor agent, then implements the task of add node information. The unregistry is to implement the task of delete node information.

**Information agent:** Corresponding to information resources. This kind of agent can obtain the parameter of the resources, the domain of the resources, and so on.

**Computational agent:** Corresponding to computational resources. This kind of agent can monitor CPU load, computational speed, total and available memory and some other host related parameters.

**Storage agent:** Corresponding to storage resources. This kind of agent can measure the typical I/O parameters such as read/write throughput, access time, etc, for a mass data storage system.

**Semantic agent:** Corresponding to semantic resources. This kind of agent aims to find precise answers to natural language questions within large document collections.

Mobile agents can move to multiple Grid resources by the position information. In stated time, system will gather the grid resource information which brought by mobile agent, and commit the corresponding status information to the portal. The unreturnable mobile agent will be retracted; user agent will be destroyed at last.

4. Conclusions and Future Work

In this paper, we have presented a multi-layer Grid portal architecture that includes mobile agent layer. The key idea is to introduce mobile agent to monitor and gain the information of Grid resources. It can reduce the network load and get the information of the Grid resources in time. Work remains to further integrate other fields by employing the multi-layer architecture and to evaluate the system according to standards.

5. REFERENCES