MOBILE AGENT BASED AUTOMATED DEPLOYMENT OF RESOURCE MONITORING SERVICE IN GRID

Valliymmai.C, Thamarai Selvi.S
Department of Computer Technology, MIT, Anna University, Chennai
{cva, stselvi}@annauniv.edu

ABSTRACT
This paper discusses a novel approach for automated deployment of resource monitoring service for job submission in grid environment. Grid computing is used for solving large scale problems which are complex. Monitoring becomes a crucial model in Grid which is used for scheduling, fault detection, accounting, etc. Job monitoring is required because user has no direct control over the job when it is submitted to a remote node. Monitoring requires services to be deployed on all the nodes in a grid environment to predict the resource availability. It is difficult to deploy the monitoring service manually in geographically distributed nodes. Hence a need for automated deployment arises. Our approach takes less time to deploy when compared to the manual deployment. The mobile agents do the automated deployment with minimum deployment time which utilizes minimum bandwidth in turn reduces the network load.

Keywords: mobile agents, resource monitoring, job monitoring

1 INTRODUCTION
A grid is a collection of heterogeneous, widely distributed resources to create a virtual organization. Grid computing helps in coordinating the resource sharing and also problem solving in dynamic, multi institutional organizations [2], [14], [16]. Grid monitoring involves collection of resource status information and providing this information which serves for system management, faults detection and performance optimization [1]. Grid monitoring is essential for efficient management and for locating failed nodes thereby ensuring that the system can run smoothly [11]. Due to dynamicity of grid environment, resources join and leave dynamically, hence monitoring becomes an essential task [3]. The available grid environment still does not support all functionalities and services like resource brokering, fault management, automatic software installation and configuration, and trust based access control [10].

A mobile agent can be viewed as an object that is capable of migrating in a network and performing task in nodes. Since logic moves near data, network load is reduced and less bandwidth is used [1]. Mobile agents are used in a variety of applications like resource monitoring, network monitoring, and copying files from remote machines, etc. In addition to the characteristics of a simple agent [12], a mobile agent has some additional characteristics like: mobility, durative, off line computing, etc. The advantages [15] of using mobile agents such as since computational unit is closer to the data, network traffic is reduced; Mobile agents reduce the usage of network bandwidth; Mobile agents can migrate themselves according to the environment; Rule based migration policy prevents blindness of resource access.

Resource monitoring involves collecting information pertaining to the system resource usage which is needed for improving the performance of the system. Network monitoring which involves monitoring the bandwidth, latency, etc is needed for better utilization of the resources [22]. Job monitoring enables to monitor the progress of the jobs submitted. Whenever a job is submitted to a node, it undergoes the following states, submitted (job that entered the Grid), waiting (while resource discovery), ready (when job transferred to the selected node), scheduled (when waiting in a local queue), running (when job is actually running), and done (when completed successfully) or aborted (when terminated abnormally) [1], [7]. System configuration is a major source of error in deployment and its management dominates system administration costs. Automation of deployment for monitoring system enables improved correctness and speed [4]. The dynamicity of grid environment raises a need for monitoring the grid. As new nodes join the grid, the resources should be monitored for efficient usage of resources. Our proposed mobile agent based automated deployment avoids the maintenance costs and human errors occurring during deployment.

The well known tool Globus [17] is used to set up the grid environment, but still it focus on low-level protocols. Due to the complex and large scale grid environment with different types of resources, there
is a need of adapting the technique for code and data mobility which is known as Mobile Agent Technology to support the features of dynamic work load balancing, reconfiguration, reliability, monitoring and management [10], [23].

2 RELATED WORK

Monitoring and Discovery Services (MDS) is used to describe and monitor services, resources, etc [18]. It is based on Lightweight Directory Access Protocol (LDAP) which does not support complex querying. In MDS, the data is not exactly latest one. Hence maintaining the resource information in a database along with a timestamp facilitates for obtaining the latest data. The Ganglia [9] presents a distributed monitoring system which has a hierarchical design. It uses a multicast-based listen/announce protocol within clusters. It maintains a tree of point-to-point connections to federate clusters and aggregate their state. Also it satisfies the requirement of monitoring in multi-clusters. When a new cluster needs to be monitored, Ganglia should be manually installed, nodes should be configured, source codes should be compiled and make command should be executed. The effort required for manual deployment becomes considerable in case of large-scale clusters [6], [13]. The present information and monitoring systems are scaled to Grid level to support the necessary checkpoints and migration. Mobile agent technology is very much flexible to support the run-time mobility through push and pull interaction models. The MA-GMA [5] is based on GMA with a mobile agent-based collecting module and a cache mechanism. GMA facilitates development of high performance monitoring middleware [8]. It is designed in accordance with OGSA standard with mobile agent’s characteristics. The mobile agents are generated in the mobile agent factory and the mobile agent manager manages the agents. The infrastructure needed for agents’ transferring and receiving in the Grid is provided by the mobile agent manager. MAGDA is a mobile agent based platform for grid environment to support the dynamic load balancing in distributed applications [10]. And the characteristics, persistence, cloning and migration of the mobile agents improve the reliability through replication. In present scenario, many researchers focus on agent based model [20] and importance of basic services such as load balancing [23], fault management [24] and service discovery [21]. Monitoring System capable of Rapid and Automated Deployment (MSRAD) utilizes peer to peer protocols for automated deployment [4]. In this paper we have proposed a method which uses mobile agents for automated deployment. Since mobile agents are capable of operating even without active connections between nodes, they are not affected by network failures. Further mobile agents reduces network load.

3 THE PROPOSED ARCHITECTURE

The proposed architecture for an agent based and Automated Deployment of job monitoring service is shown in the Figure 1. The main components of the architecture are as follows:

![Architecture of agent based automated deployment](image)

**Figure 1.** Architecture of agent based automated deployment
Deployment Agent: an agent which contains code for deployment of the services. It resides in the head node. On request from a newly arrived node, it migrates to the node and executes.

Registration Node: A node which maintains a database of the nodes that arrive and also it sends the IP address of the head node to the newly arrived node.

Probe: A script is deployed in the nodes. On invoking the script it collects the resource metrics of the nodes periodically and stores in the head node in the form of XML.

Collector Service: The collector service collects the resource metric from the head node and stores it in a database in the head node along with a time stamp.

Database: A database that stores the resource information gathered from every node which will be used for monitoring and scheduling.

Selector Service: The selector service searches the database for a node that best suits the user’s job requirements.

Job monitor: This module performs credential delegations and submits the job to the selected node for execution. It also updates the status of the job periodically.

A mobile agent is composed of code and data which migrates to other nodes and executes there in the node to which it migrates. The mobile agents exploit the basic communication protocols defined within IBM Aglets Workbench [19] for agent migration and to dispatch messages from one node to another node. The head node contains the services to be deployed and the deployment agent. The registration node maintains a repository of IP addresses of the nodes in which the monitoring service is deployed and also the IP address of the head node. Mobile agents are used by the user for getting the IP address of the head node from the registration node and for deploying the services. For automated deployment of service, mobile agents are used to send request to the registration node on its arrival and retrieve the IP address of the head node to the newly arrived node. A mobile agent from the new node migrates to the registration node, collects the IP address of the head node and migrates back to the sender node. Using the IP address of the head node, the newly arrived node requests the head node for deployment of the services. The deployment agent migrates to the head node containing the services and executes thereby deploying the services.

4 IMPLEMENTATION

The implementation of the proposed approach for an automated deployment of the services to monitor the job has two phases. The first phase involves creating services that are used for automated deployment.

One service called collector service is created for collecting the resource metrics from the system and another service called selector service is used to select a suitable node for the submission of job. The collector service includes collecting resource metrics like CPU speed, free memory, bandwidth and latency from the nodes in the cluster. The collected data is then stored in the database along with a timestamp. Timestamp helps in validating the monitored data. The selector service selects a suitable node by searching the database based on some constraint. The constraint used here is, select a set of nodes having high free memory, from this set a node having more CPU speed, low latency and high bandwidth is chosen for job submission.

![Figure2. Automated deployment using mobile agents](image-url)
The second phase involves automating the service deployment. The process of automated deployment of the service to a new node is shown in Figure 2. Whenever a new node joins the grid, it registers itself to the registration node. The registration is done by sending the IP address. The registration node maintains a database of the IP address. Now the registration node sends the IP address of the node where the service is located i.e. in the head node. After getting the IP address of the head node the newly joined node invokes the deployment agent in the head node. The agent deploys the service requested in the new node. The deployment agent has the details of which files to be transferred and what commands to be run for the deployment. These services are used for selecting a node to which a job can be submitted. GRAM services are used which helps in secure job submission to various types of schedulers. Job monitoring module periodically updates the job status and after completion of job the result is reported to the user.

5 RESULTS AND DISCUSSION

The time taken for the mobile agents to deploy the service in a new node is calculated as follows:

Let \( t(DT) \) be the time taken for automated deployment.

\[
\begin{align*}
\text{t(DT)} &= \text{t(\alpha)} + \text{t(\beta)} \quad (1)
\end{align*}
\]

Where

\( \text{t(\alpha)} \) is the time taken for the deployment agent to migrate to a node.

\( \text{t(\beta)} \) is the time taken for deployment.

From the Figure 3, it is evident that mobile agent based automated deployment consumes less time than the script based approach in Automated Deployment of Monitoring Service (ADMS). If the dependency packages for Ganglia are downloaded, the time taken to install Ganglia is equal to running four commands which completes the installation. ADMS has to install Component Auto-deploy Proxy (CAP) and cluster and thereby it takes more time than the mobile based approach.

A simulation based test is adopted to verify the efficiency of the proposed method. It is seen that the time taken for manual deployment increases as the number of nodes increases. Since the response time of mobile agent is less, the time taken for automated deployment using mobile agent does not increase proportionately with the increase in the number of nodes.

6 CONCLUSION AND FUTURE WORK

In this paper, we have specified the importance of monitoring and automated deployment of monitoring service in Grid environment. The proposed approach enables automated deployment in a grid environment with the help of mobile agents. Mobile agents are utilized in the automated deployment process to facilitate error free deployment since mobile agents are not affected by network failures and the time taken for deployment is also less when compared to manual deployment. The automated deployment of monitoring service exploits the plan of selecting suitable node which ensures the efficient utilization of grid resources. Our future work includes extending the architecture which supports the resubmission of failed jobs and protection of mobile agents form malicious hosts to improve the reliability of the automated deployment using mobile agents. The monitoring service can be invoked to select another suitable node for job submission and the failed job can be resubmitted in the selected node, and also reporting the status of the job given to the user.

7 REFERENCES


