Development of Resource/Commander Agent

For AgentTeamwork Grid Computing Middleware

Enoch Mak
Dept. of Computing & Software Systems
University of Washington, Bothell

Prof. Munehiro Fukuda, Ph.D.
Dept. of Computing & Software Systems
University of Washington, Bothell

Introduction

AgentTeamwork is a grid-computing middleware system which dispatches a collection of mobile agents to coordinates an execution of a user application over remote computers in a decentralized manner. AgentTeamwork do not need a powerful central server like the traditional centralized grid-computing middleware. The ultimate focus is to maintain a high availability and dynamic balancing of distributed computing resources to a parallel-computing job.

Mobile Agents

AgentTeamwork consists of four types of mobile agents:

Commander agent : Coordinates the work of other agents.

Resource agent : Dynamically allocates distributed computing resource to a parallel computing job.

Sentinel agent : Monitors and takes snapshot of the execution of user application at a different machine.

Bookkeeper agent : Keeps the periodical execution snapshot from sentinel agent.

Resource Agent Tasks

Resource Agent task sequence:

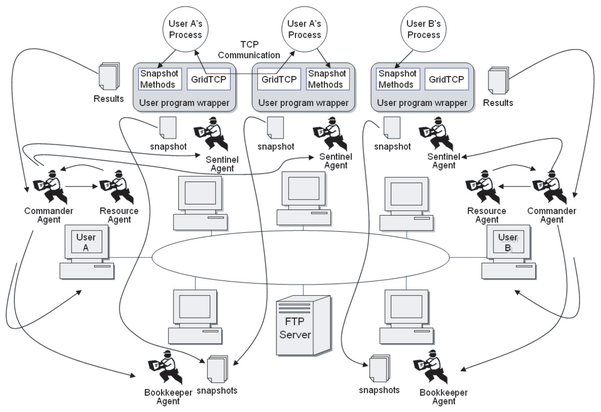
1. Access to a central FTP server
2. Download new resource XML files
3. Maintains the XML files in local database
4. Query the local database with the user job requirements
5. Return a list of computing resource to Commander Agent
6. Perform periodic remote probing

FTP Server

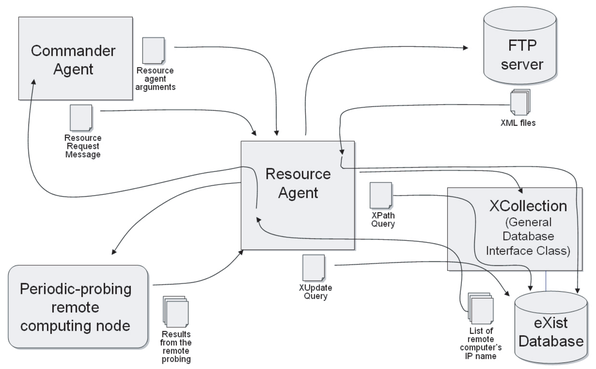
The purpose of the FTP server:

- To store a collection of XML-based user resource files and AgentTeamwork software kit.
- Does not used as a central cycle server for job submission and coordination.

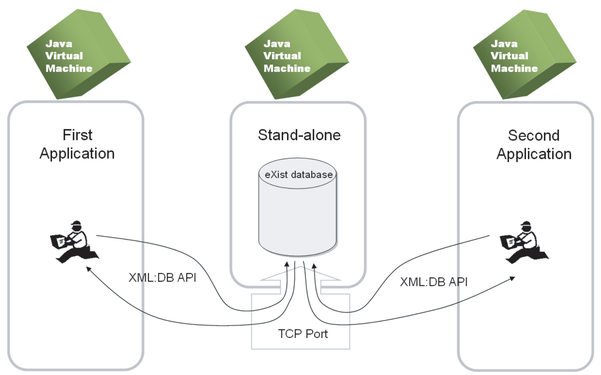
AgentTeamwork System Overview



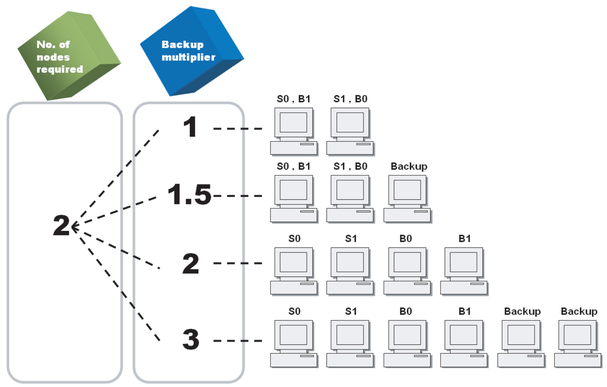
Resource Agent Workflow



eXist Database Server



Computing Resource Distribution



Implementation Time Schedule

Acknowledgement

This project was made possible by National Science Foundation (NSF) and the Distributed Systems Laboratory (DSL) of the Computing and Software Systems (CSS) Department at University of Washington, Bothell (UWB). Thanks are extended to the guidance and support of Prof. Munehiro Fukuda, Ph.D.

For further information: http://depts.washington.edu/dslab/AgentTeamwork

Contact : enochmak@u.washington.edu, mfukuda@u.washington.edu

National Science Foundation
WHERE DISCOVERIES BEGIN

eXist Database

eXist is a open source native XML database system, it maintains the information of all remote computers.

Mulitipler resource agents may run locally in parallel when the user invoke multiple applications at the same time. Thus, the eXist database is deployed as a stand-alone server to meet this requirement. The stand-alone database server allow resource agents to access through a TCP port with standard XML:DB API.

Remote Probing Test

Resource agent periodically checks the status of all remote computing nodes which are enumerated in the local database.

Five pieces of information will be collected:

- Operating System type
- Available memory
- Free disk space
- Percentage of CPU idel
- Bandwidth between Resource agent and remote computers

Conclusion

- The first version of resource agent was developed for the AgentTeamwork project
- Commander agent was enhanced in order to cooperate with Resource agent
- AgentTeamwork is now able to perform a complete sequence of job execution:
  1. an application is submitted,
  2. a commander is spawned and send a resource query to the resource agent,
  3. the resource accesses its local eXist DB and returns a list of computers,
  4. a commander dispatches sentinels and bookkeepers to launch and to monitor the application,
  5. the commander receives the final computation results.

Future Work

- Implement the bandwidth test server with Java
- Perform bandwidth test between remote computers
- Enhance resource agent to be resumable by bookkeeper agent
- Provide a new list of idle computers so that agents can mirgrate to a lighter loaded machine