Abstract: Hierarchical distributed architecture was presented for mobile grid based on overlay network, and the special needs of mobile grid middleware were analyzed. Based on these analyses, modular framework of mobile node middleware was designed, as well as that of mobile administrant node. The function of each module and their interactions are described respectively. The proposed mobile grid middleware combines the flexibility of mobile devices with the processing power of the grid. It allows mobile devices to use grid resource and enables the execution of resource-intensive applications on the resource-constrained devices. The proposed middleware also solves intermittent connection problem of mobile device which was taken as resource provider or resource consumer. Finally, it was pointed out the future works about mobile grid.

Keywords: mobile grid; architecture; middleware; intermittent connection

1 Introduction

With the technical development of grid and widely used of mobile devices, researchers start to consider grid support for mobile devices, resulting in the concept of mobile grid [1]. Mobile grid is cross-area of grid computing and mobile computing, which becomes a new hotspot. Generally accepted view on mobile grid is: it is inheritor of grid with the additional feature of supporting mobile users and resources in a seamless, transparent, secure and efficient way [2].

1.1 Study object and architecture of mobile grid

Traditional grid mostly consists of large computing devices such as high-performance computers, super servers and workstations. Based on traditional grid, mobile grid also includes all types of wireless devices such as cellular phone, PDA, portable computers, devices on vehicles, laptop, together with wired ordinary nodes such as PC. The performance of PC is not as good as that of high-performance computers, but usually superior than that of mobile device. If mobile device can be integrated into grid, this type of fixed resources should also have a place in grid. Therefore, the grid size will expand rapidly due to the large number of mobile and fixed nodes. Under this condition, a centralized or hierarchical organization of resources would make the single point of failure, server bottlenecks, scalability issues become more difficult to resolve. Therefore, it is necessary to consider the mobile grid architecture from distributed perspective.

Overlay network [3-4] provides ideas for the design of mobile grid. Overlay network is good at hiding the dynamic and heterogeneity of underlying physical network, especially suitable for mobile grid environment. Figure 1 shows mobile grid architecture based on overlay network. Proposed architecture adopts hierarchical and distributed resource management.

The entity of mobile grid includes resource nodes and administrant nodes. Resource nodes are various types of computing devices, which are divided into multiple sub-networks based on certain strategy. These sub-networks form underlying resource network and each sub-network is controlled by an administrant node. All administrant nodes are connected by DHT overlay network, composing upper administrant network. It is necessary to point out that the management of mobile devices is based on existing wireless network infrastructure such as wireless LAN and cellular network. Mobile administrant node is deployed on wireless access points. Its function is different from that of fixed administrant node. This paper focuses on the middleware design of mobile administrant node and mobile device.

1.2 The Roles of Mobile Device
1.2.1 Interface to Mobile Grid

It is difficult for user to perform complex task with mobile device due to its limited capabilities on computing, storage and communication. In mobile grid environment, user can share grid resources by taking mobile device as grid interface. As mobile devices often disconnect from network and mobile network itself is unreliable, it is not advisable for mobile device to interact with grid sites directly. Under the above architecture model, administator node interacts with other grid entities on behalf of the interests of a group of mobile devices, performing resource discovery, task submission, monitoring, cancellation, etc [5].

1.2.2 Grid Resource Provider

Mobile device resource can also be added to current grid, extending existing resources of grid. Although single mobile device has very limited resource, the amount of resources provided by a huge number of mobile devices is considerable. K*Grid project [6] have achieved some results in using idle resources of mobile devices. In order to achieve this function of mobile device, mobile administrator node will perform resource management, hiding platform heterogeneity, task decomposition, task scheduling, fault tolerance, quality of service and other functions.

2 Challenges of Mobile Grid Middleware

In mobile grid environment, middleware has its unique characteristics compared to that of traditional grid. One should consider following factors when designs mobile grid middleware.

2.1 High Degree of Heterogeneity

Compared with traditional grid, mobile grid includes more types of resources which vary dramatically in CPU speed, storage capacity, and network bandwidth and connection stability. Especially in mobile devices, there is no uniform industry standard and there are different operating systems and network interfaces, such as cellular, radio, Wi-Fi and Bluetooth.

2.2 Context Awareness

Mobile grid integrates grid computing and mobile computing. In order to implement distributed applications, one must consider the context of mobile device, such as network bandwidth, battery power, location, etc. In the middleware framework, context-sensitive is an important component which is responsible for collection, storage and management of context information, providing interface for external objects to access. External objects can obtain context information by querying actively. More generally, middleware will inform external objects with the changes about the context information.

2.3 Limited Resource of Mobile Device

There are diverse devices in mobile grid. A variety of resource-constrained devices such as cellular phones, PDA and embedded computing devices are included in mobile grid. These terminals have limited processing capabilities, so middleware must be lightweight and cannot take up too much system resources in order to support mobile distributed applications.

2.4 Intermittent Connectivity of Mobile Devices

Wireless communication and unstable performance of mobile device bring great challenges for the integration of mobile device into grid system. Wireless link failure, battery power exhaustion, and abnormal shutdown or out of coverage may lead to the connection between mobile device and grid interrupted. And that this interruption is not expected. Application will be severely affected due to the interruption if mobile device which is performing grid task or requesting for grid resource at this time. Therefore, it should be given full consideration to the intermittent connectivity of mobile device.

2.5 Security

Security is not unique requirement of mobile grid. However, security issue becomes more complex because of wireless communication. How can we ensure security of communication process and user’s privacy as well as how to control excessive consumption of safe computing tasks in resource-constrained devices are important issues need to be addressed?

3 Middleware Framework of Mobile Grid

The design of mobile grid middleware combines characteristics of grid computing and mobile computing [7], giving full consideration to its special needs. Figure 2 shows middleware framework of mobile grid.

3.1 Middleware architecture for mobile node

Mobile node middleware consists of monitor, device agent module, cache module, personal information module.

Monitor resides on each mobile node in mobile grid. It is responsible for collecting status information of mobile node, including connection status, CPU load, remaining battery power and available memory and storage space. The implement of monitor can learn MoCA architecture [8]. As a daemon running on each mobile node, MoCA is responsible for collecting status information, such as RF signal strength, energy level, CPU usage and storage capacity. Personal information is used to store user information or mobile device safety data, which are needed during the certification or before invoking a particular grid service.

Cache module stores interface code and service
states so as to avoid requesting same information every time. When submitted job by mobile node was completed, the returned result is stored in cache module.

Device agent is a daemon which is responsible for agent discovery, local environment initialization or terminating communication between mobile device and its agent. It can use well-known service discovery protocol such as UPnP to search vicinal agent (namely mobile administgrant node) in its surrounding network environment.

3.2 Middleware architecture of mobile administgrant node

Context listener communicates with monitor on mobile node and collects its dynamic information periodically. Then it informs information server to update database. Compared with traditional grid, in addition to node availability, bandwidth and load, the dynamic information also include connection state, battery power, whether in coverage, etc. Mobile node reports its availability status to context listener actively in the following cases: turn on/off; link status from disconnection to normal; moving to a new administrant domain. After receiving the report, context listener informs information server to update the availability of mobile node. During information collection, if a mobile node disconnects (for example, abnormal shutdown, energy exhaustion, interruption of communication link, host sleep mode, host moving out of administrant domain), context listener will start a timer. If mobile node does not restore connection after the set time, context listener will modify its status unavailable.

Information service provides directory service of local resources and some off-site resources. It is adopted distributed resource management in mobile grid. Each administrant node manages all local resources and some off-site resources. All the administrant nodes constitutes overlay network through DHT technology which manages all grid resources. Therefore, in addition to local nodes information, information service also saves a number of adjacent (adjacent logically) nodes information. Static information of local resource such as IP address, operating system, hardware architecture, CPU type, CPU speed, memory size and disk space are obtained when node registered. Dynamic information of local resource are collected periodically by context listener. Information service can use LDAP which is a protocol widely used in many grid projects.

Job manager handles application requests from mobile node. In this case, user takes mobile device as grid interface to complete complex tasks. Job manager consists of controller, job agent and collector. Controller sorts the received requests and assigns a unique identifier to each application. Application request contains not only resource requirement but also indicative information defined by mobile user on how to deal with mobile node disconnection. Then controller sends request to resource selector which returns a list later. There is some available resources information for certain applications in the list. Job manager chooses proper resources for application based on pre-defined strategy. After assigning each task to selected resource, job manager creates a job agent for the application. Job agent will test connection state of the mobile node periodically until the application is completed. The results sent back by resources are saved in collector and then returned to user. During the execution of application, controller should monitor the running of tasks, collect status information and store them in the checkpoint file of collector. Interactions between job manager and other components are shown in Figure 3.

Job agent will conclude disconnection if it has not received response from mobile node in a given interval. Then job agent analyzes application features and user-defined indicative information, and decides how to deal with disconnection. Once disconnection occurs, job agent can perform three kinds of operations on application: continue; suspend and terminate. If application does not has any dependence on mobile user, that is, applica-
tion does not require data from user during execution, user can define that application continue running after disconnection occurs. The result of application is stored in collector first. At this time if connection has been restored, the result is sent back to user at once, and job manager write-off job-related data. Otherwise, job manager saves the result until mobile device reconnects. When mobile device reconnects, application starts from the point where it is stopped. The third operation performed by job agent is to terminate application. It is applicable to above two cases and decided by user. In this case, job manager informs resources to cancel tasks and write-off relevant data. When mobile device restore connection, mobile user submits application request again.

Resource selector For resource request from mobile node, one of principles is using fixed node resource as much as possible. Therefore, after resource selector receives request from job manager, it does not access local resource directory but access off-site directory to find fixed resource. If selector can not find proper resources, it will send the request to adjacent fixed administrant node. Since topology consistency of logical neighbors and physical neighbors is taken into account on the construction of overlay network, resource requesters and resource providers are always physically adjacent which effectively reduces network traffic and communication delays.

Application Schedule module performs task schedule on a group of mobile devices. The purpose of this module is to make full use of idle resources of a large number of mobile devices to complete intensive computing similar to seti@home project. In mobile grid schedule, job model adopts divisible load which is simple but can be used to characterize a lot of large-scale application and suitable for mobile grid environment. When application schedule module receives application request from external administrant node, job divider first splits the job into multiple independent sub-tasks based on application’s feature. Then job scheduler invokes information service to know current available resources and their dynamic performance. Task schedule on mobile resources should give full consideration to task characteristics. For example, nodes with sufficient energy should be given priority if tasks need long execution time. Scheduling communication-intensive tasks should choose well-connective nodes. Application may also specify maximum and minimum number of resources. Job scheduler selects proper resources for each sub-task after a trade-off among these factors. At last job assigner sends sub-tasks to selected resources and collects the result of each sub-task. The results are integrated and sent to requesting node.

Throughout scheduling process, mobile device may disconnects from network due to various reasons, with the result that sub-task can not be completed properly. Therefore scheduler should also consider proper strategy in response to intermittent of mobile resource. Possible strategies include re-scheduling, task migration, task replication and so on.

4 Conclusions and future work

Integration of mobile devices into grid system relies on existing wireless network infrastructure. Mobile administrant node located on AP is on behalf of a group of mobile devices to interact with grid, which effectively shields dynamic and heterogeneity of mobile node.

Traditional grid middleware and mobile computing middleware has its own application environment and can not be directly deployed in mobile grid. The requirements of mobile grid middleware are analyzed in this paper and then middleware architecture of mobile node & administrant node are presented. This paper also gives a detailed analysis of basic function components required for mobile device to play two roles—resource consumer and resource provider.

There are a lot of research work for mobile grid such as construction of overlay network, resource discovery, resource description, schedule algorithms, security and fault-tolerance. Under the condition that grid computing and mobile computing are relatively mature, carrying out research on mobile grid is possible and necessary.

Acknowledgement

This reserach was supported by by Doctoral Fund of Ministry of Education of China. We would like to thank Professor YU Zhen-Wei for his instruction and comment on the work.

References


