Journal of Grid Computing (2004) 2: 193–206

Grid Computing in China

Guangwen Yang[1], Hai Jin[2], Minglu Li[3], Nong Xiao[4], Wei Li[5], Zhaohui Wu[6], Yongwei Wu[1] and Feilong Tang[3]
[1]*Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China*
[2]*School of Computer, Huazhong University of Science and Technology, Wuhan 430074, China*
[3]*Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030, China E-mail: li-ml@cs.sjtu.edu.cn*
[4]*Key Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, China*
[5]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China*
[6]*College of Computer Science, Zhejiang University, Hangzhou 310027, China*

Key words: ChinaGrid, China National Grid, Grid, NSFC Grid, ShanghaiGrid

Abstract

Grid computing presents a new trend to distributed computation and Internet applications, which can construct a virtual single image of heterogeneous resources, provide uniform application interface and integrate widespread computational resources into super, ubiquitous and transparent aggregation. In the adoption of Grid computing, China, who is facing more resource heterogeneity and other specific demands, has put much effort on both research and practical utilization. In this paper, we introduce the major China Grid research projects and their perspective applications. First we give the overview of the four government-sponsored programs in Grid, namely the China National Grid, ChinaGrid, NSFC Grid, and ShanghaiGrid. Then we present six representative ongoing Grid systems in details, which are categorized into Grid middleware and Grid application. This paper provides the general picture of Grid computing in China, and shows the great efforts, devotion and confidence in China to use Grid technology to boost the society, economics and scientific research.

1. Introduction

Grid computing, which presents quality-guaranteed virtual single image of geographically distributed resources such as processing, storage, and instruments, provides promising solutions to contemporary users who want to effectively share and collaborate among themselves in distributed and self-governing environments.

In the adoption of Grid computing, China, who is facing more heterogeneity and specific demands, has launched four programs, which are China National Grid [2], ChinaGrid [1], NSFC Grid [13], and ShanghaiGrid [10], with different emphasis to adopt Grid computing to China's practices and boost its further advance. The goal of China National Grid is to build a distributed high performance environment to support various pertinent applications. By aggregating existing dispersed resources through China Education and Research Network (CERNET), ChinaGrid aims to build the largest education and research platform with 100 universities across the country to collaborate on research, science and education projects. Constructing a virtual research and experiment environment, which permits wide-area collaboration, is NSFC Grid's primary concern. As a regional Grid, ShanghaiGrid concentrates on transportation traffics. Various applications are supported in these programs/projects, including weather forecasting, petroleum reservoir simulation, high-energy physics, numerical wind tunnel simulation, ship structure analysis, Alpha Magnetic Spectrometer data processing, traffic control and guidance, etc.

In building Grids, middleware is a core building block to support cooperation among machines and to provide functionalities such as resource management, monitoring, authentication and authorization. Although Globus [14] has provided an open source software toolkit for constructing Grids, more middleware is still needed (maybe built on top of Globus) for ease of building Grid infrastructure and applications.

The rest of the paper is organized as follows. In Section 2, we will introduce the four key Grid programs in China. In Section 3, we describe some middleware software built by the universities and research institutes in China. Section 4 lists some applications of Grid computing in China, and Section 5 concludes the paper.

2. Key Grid Programs in China

For presenting a globe view of Grid computing in China, in this section we will introduce four ongoing Grid programs which are the largest Grid plans sponsored by Chinese government. We will describe the initiators, goals, participants, contents, and statuses of each program. With the four programs well done, Grid computing will contribute a lot in China, and as a result advancing China's developments in society, economics, and research.

2.1. *China National Grid*

The China National Grid Program (CNGrid) [2] is supported by the National High-Tech R&D Program (the 863 program) of China. It is a 4-year plan comprising about 20 sub-projects. The goal of this project is to construct a distributed high performance environment to support various relevant applications.

To achieve this goal, four kinds of tasks are carried out at universities, research institutes, and companies: First, a supercomputer providing centralized Grid services is to be developed for major Grid nodes. Second, several high performance computing centers is being established across the country, on top of which the Grid infrastructure is constructed. The third task is to develop common Grid software supporting applications. Finally, some Grid applications run on CNGrid to test the system performance. Up to now, there are totally nine high performance computing centers having been built (located in Beijing, Shanghai, Hong Kong, etc.), interconnected by network backbones (CERNET, which is the China Education and Research Network, and CSTNet). Hosts in these centers include Dawning 2000/3000, Galaxy 3, Sunway, Tsinghua Tongfang PC-Cluster, HKU Gideon 300 cluster, etc. Supporting software, such as resource and user management, authentication, monitoring, Grid user interface, is under development. The potential Grid applications involve weather forecasting, petroleum reservoir simulation, numerical wind tunnel simulation, and ship structure analysis.

Different from the other three key Grid programs, this program is particularly focusing on industrial applications. It is initiated as a key project in the 863 program with a 100 million RMB (about 12 million USD) government fund, and is planned to obtain $2 \sim 3$ times of investment from industry more than the government fund.

2.2. *ChinaGrid*

ChinaGrid [1] is a long term project with more than 500 M RMB (about 60 M USD dollars) fund supported by the Ministry of Education of China. It is one of the world's largest implementations of Grid computing. Its goal is to enable 100 universities across the country to collaborate on research, scientific and education projects, based on Grid technology and CERNET (China Education and Research Network), so as to make miscellaneous distributed resources available where and when they are needed.

ChinaGrid mainly focuses on three main tasks: Construct more than 30 high performance computing sites (cluster mainly) with 15 TeraFLOPS peak performance totally; Develop Grid computing middleware and deploy it to the universities so as to provide gigantic computation power to the desktop of end users; and Develop and deploy several specific computing and service Grid applications, such as bioinformatics Grid, image processing Grid, computational fluid dynamics (CFD) Grid, remote education Grid, massive data processing Grid.

2.3. *NSFC Grid*

Network-based scientific activity research environment [13] is a four-year program starting from 2003, initiated by National Science Foundation of China with a budget of RMB 30 millions (about 3.7 million USD dollars). It directs at building a virtual environment for research and scientific experiments. It will allow regional or even global collaboration among labs and scientific instruments, which will consequently change contemporary style of research and greatly improve the efficiency. By integrating geographically distributed computing utilities, storage devices and instruments through network technologies, NSFC Grid enables far-flung sharing, effective aggregating, and fully releasing of computing, data, and service resources, therefore creates universal infrastructures for large-scale computing and data process.

Fundamental theoretical science problems, computer architecture and core techniques, and synthetic experiment platform in network computing environment are main concerns of NSFC Grid. It is planned to build up experiment systems for high-energy physics, atmosphere information, and other problems based on network computing. The research of fundamental science theories in network computing environment can help scientists profoundly understand the essential principles of network computing and promote innovations, hence greatly improving the capacities of data processing. Relying on NSFCnet and Next Generation Internet Pilot project under construction, connecting important science research resources, including computing Grid and data Grid, providing collaboration environment based on network for cross-region and cross-discipline research activities, and supporting research activities in classical application fields, synthetic network experiment platform will verify the fundamental science theory and critical technologies in network computing environment: architecture and standards, application-support middleware, resource-support middleware, security-support environment, application-orientated resolving system. High-energy physics, atmosphere, and bioinformatics biology are the three application cases built on the synthetic experiment platform.

When all these done, the program will provide virtual environment for scientific activities in important fields, hence greatly influencing the developments of Chinese society and economics, scientific research, style of living and work.

2.4. *ShanghaiGrid*

ShanghaiGrid [10] is a long term plan sponsored by Science and Technology Commission of Shanghai Municipality (STCSM) with 50 million RMB funds (about 6 million USD dollars) from the government and others. It aims at constructing metropolis-area information Grid infrastructure and establishing an open standard for widespread upper-layer applications from both research communities and official departments.

The ShanghaiGrid project is based on the current four major computational aggregations and networks in Shanghai, i.e. the CHINANET (public internet backbone built by China Telecom), the SHERNET (Shanghai Education and Research Network), Shanghai Supercomputing Center, and campus networks in Shanghai Jiao Tong University, Tongji University and Shanghai University, and is planned to enable the heterogeneous and distributed resources to collaborate into an information fountain and computation environment for research, education and metropolis management applications, seamlessly and transparently. To achieve the goal, the project consists of four interdependent sub-projects: the research and investigation on requirements, protocols and standards of information Grid infrastructure, the development of system software and establishment of major Grid nodes, the development of decentralized virtual research platform, and research on metropolis Grid applications (mainly the Shanghai traffic-congestion control and guidance application). Relying on the growing network infrastructure and abundant scientific research resources, the ShanghaiGrid project will construct the first metropolis-area information Grid to provide tremendous data manipulations and ubiquitous information services for variety of organizations. Especially, the traffic-congestion control and guidance application is planned to take advantage of Grid computing to integrate traffic data collection, traffic monitoring and supervising, taxi information supply and traveler guidance, in order to make the traffic system run more efficient and people easier to travel within the city.

Table 1 gives a brief overview of the above four Grid programs.

3. Middleware

Middleware is a core component to effectively construct Grid computing system. Great efforts have been made to develop scalable, secure, and highly available Grid platforms on top of local operating systems, which transparently shield the heterogeneities and dynamic behaviors of participants. Four important platforms in China have been developed or under construction.

3.1. *T.G*

T.G is one of the earliest efforts in Grid computing in China, which is supported by China National High-Tech R&D Program and developed by the Grid

Table 1. Brief information of four key Grid programs in China.

Project name	Funding	Intent	Dates	Developers
China National Grid	100 million RMB from National High-Tech R&D Program of China	Construct a distributed high performance environment to support various applications	2002–2006	More than 20 universities, institutions and enterprises in mainland China and Hong Kong
ChinaGrid	500 million RMB from Ministry of Education of China	Enable 100 universities across the country to collaborate on research, scientific and education projects	2002–2006	More than 30 universities in mainland China
NSFC Grid	30 million RMB from National Science Foundation of China	Build a virtual environment for research and scientific experiments	2003–2007	More than 12 universities and institutions in mainland China
ShanghaiGrid	50 million RMB from Science and Technology Commission of Shanghai Municipality and others	The first metropolis-area Information Grid in China	2003–2005	Several universities, institutions and application departments in Shanghai, China

Computing Group of Tsinghua University since 2000. The initial purpose of T.G is to effectively aggregate various computing resources on campus of Tsinghua University into a powerful computational environment and provide unified and transparent interface for users.

Main focuses of T.G include resource management, task scheduling, and parallel programming model.

Resource Management

As to resource management, it presents a globally uniform hierarchical resource model. Resource management consists of both resource abstraction and resource organization. Although computing resources distributed over campus are various and heterogeneous in many aspects, some attributes can be abstracted for their uniform description. These parameters contain two parts: one is static, which includes relatively fixed information such as operating system, architecture, network IP address, total CPU number and frequency; the other part may vary dynamically, which includes computing capacity, current CPU load, available memory size, etc. By this means, each computing resource can be represented by a combinational set of static and dynamic parameters.

Based on the uniform resource description, T.G organizes resources into a hierarchical tree-like structure (Figure 1), in which each node has a father (root node excluded), several siblings and children. A resource node in the resource tree along with all its offspring nodes constitutes a sub resource tree with the node as leader. T.G adopts SciMark2, a set of benchmark designed for scientific computation to measure the computing capacity of each resource node. The root of each resource tree or subtree records the whole computing capacity of all the nodes in this tree. A node communicates with its father periodically to report the current capacity status of the subtree leaded by it.

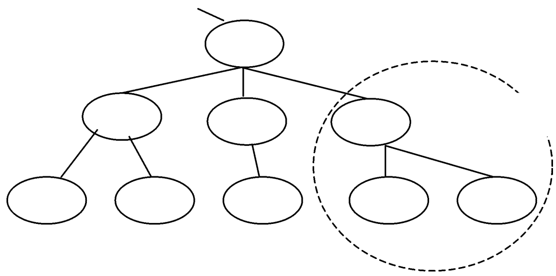

Figure 1. Hierarchical tree-like resource structure.

Construction of resource tree relies on the information center in the whole resource system. It periodically retrieves newest resource tree from the root node. When a new computing node intends to join the system, it first connects to the information center, and the latter will designate a father node for this new node. And it assumes that nodes with low network communication cost reside in the same/adjacent subtree. Then the new node communicates with its father and both of them get aware of each other.

Task Scheduling

Corresponding to the hierarchical resource tree, we also organize user's task in a logical tree. User's applications supported by T.G are common with a divisible workload, namely the application consists of several atom tasks. Users can partition their applications and put those atom tasks which have communication between them in the same subtree. Only leaf nodes in the task tree are atom tasks with actual computation.

Task scheduling can be converted to find a suitable resource subtree for user's task tree. It is accomplished in two steps. First, to find a sub resource tree to meet the requirement of user's application. Second, to find qualified nodes in the subtree for each atom task contained in user's requests. When a user submits the task request to a resource node, the node first checks whether the resource sub-tree is able to satisfy the requirements of user's application. If so, it will select one qualified child node with minimum superfluous computing capacity, and redirect user's request to this child. If not, the user's request will be redirected to its father node, which performs similar operations as above. All process is enforced literally. Once the user's request is accepted by a node, the first stage is finished. Then the node performs scheduling process for each atom task contained in the task. All the operations are performed recursively until each atom task finds the appropriate running node for itself.

Programming Model

T.G also puts forward a thin kernel parallel programming model for Grid computing. The term "thin kernel" means the separation between the actual computation code (referred to as kernel) and a task partition. In traditional parallel programming model, the partition of a user's parallel tasks is hard-coded into programs. So when a programmer intends to partition a task, he has to modify the source code, re-compile and re-link into new executables. In the thin kernel programming model, a user can implement his actual application with common programming language and describe his task partition simply with a XML file. The user can adjust the running of the whole application by only changing the task partition part without modifing the kernel code. Besides, during task scheduling process, only task partition part in the application description file needs to be transferred between nodes. Only after the appropriate nodes have been selected for user's application, actual codes and data will then be transferred to these nodes for execution. So this efficiently reduces traffic amount and saves network bandwidth.

T.G also provides users with a simple interface to implement atom task in Java. The API mainly consists of three key methods. Init( ) method is responsible for initialization of atom task, including preparing and parsing necessary input parameters; kernel( ) method contains material of computation code; finalizeAtom( ) method is responsible for sending completion notification or results. When an atom task is assigned to a node, the node starts a new thread for the task and invoke the three methods in order.

We have released an initial version of T.G. With T.G, users can easily construct a computational Grid platform with their available resources, develop practical applications with the API provided and enjoy the huge power of resource pool.

We have also constructed an experimental Grid testbed in Tsinghua University, and several applications, which range from data security, bioinformatics, to material science, have been successfully deployed on the testbed. We also plan to extend T.G to deploy it in the ChinaGrid, to construct a single, virtual high performance computing platform for the users in CERNET.

3.2. *Vega*

Vega is a six-year Grid project starting from 2001, funded by R&D grants from multiple governmental agencies, which aims at constructing supercomputers, massive storage systems and other resources into a uniform Grid platform. Currently, research efforts focus on the following three layers, as depicted in Figure 2.

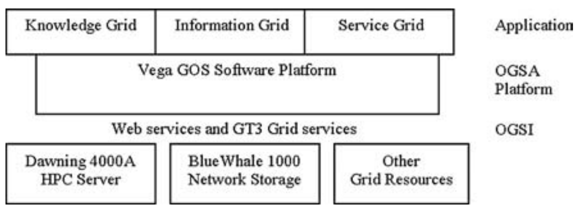

Figure 2. Vega Grid research activities.

Dawning 4000A High-Performance Servers [24]. Dawning 4000A is ICT's first attempt to build Grid-enabling servers. It is a 10 TeraFlop/s high-performance computer with an enhanced cluster architecture. Dawning 4000A comes with a hardware box called Grid Router, and enhancement to its cluster operating system to support OSGI and the Vega Grid software. Dawning 4000A is scheduled to be installed at Shanghai Supercomputer Center in June 2004, as the main server of a major node in China National Grid.

BlueWhale Network Storage Systems. ICT announced the completion/and shipment of the first network storage product, BlueWhale 1000, in October 2003. The system provides Infiniband and Gigabit Ethernet interfaces, and enables storage resource sharing at the device and the filesystem levels. The system supports up to 512 TB storage, divided into up to 256 storage devices, and each device can provide a sustained bandwidth of 120 MB/s. The system supports dynamic deployment of operating system images and applications, with a 4-minute per copy deployment time. It provides a single control point to monitor and manage the entire storage system.

Vega Grid Platform. The project has released the Vega GOS version 1.0 software, which is OSGA-compliant and is implemented in Java. The Vega GOS software encapsulates all resources as Grid services or web services, and uses GT3 as the runtime environment for Grid service. It also supports other OGSI [15, ?] features, such as GSI security, data transferring by GridFTP or GASS, GT3 GRAM. The software has been deployed on all nodes of the China National Grid, forming a cross-domain and cross-region Grid testbed. Vega GOS 2.0 and 3.0 will be released by the ends of year 2004 and 2005, respectively.

Vega Information Grid. The project has developed an information Grid framework based on web services. Two products of information Grid middleware are under test at user sites for transportation industry and for enterprise information platform applications. A main feature of Vega Information Grid is the support for change management, including resource changes and user requirement changes. Besides improving products quality and functionality, the next step is to convert this framework to take advantage of OGSA/WSRF, and to support e-governant and digital library applications.

Vega Service Grid [8]. Drawing from applications requirements in manufacturing industry and digital Olympics, the project developed a CAFISE approach, including language, model and supporting tools for adaptable application development in Grids. CAFISE includes a user-centric, business-level service composition language called VINCA (for Visual and Personalized Business-level Composition Language for Chaining Web-based Services). To facilitate the construction and execution of VINCA applications, a suite of CAFISE toolkit is developed. Based on the MASON model, the toolkit enables just-in-time construction and dynamical adaptation of service-oriented applications.

Vega Knowledge Grid [25]. Also known as China Knowledge Grid research group (VEGA-KG, http://kg.ict.ac.cn), this team has been pursuing the best solutions of three fundamental scientific issues of the next generation interconnected environment – normally organize, semantically interconnect and dynamically cluster globally distributed resources. The project has implemented an experimental knowledge Grid system, which includes a multi-dimensional knowledge space model, a knowledge operation language and a knowledge flow model for realizing effective knowledge sharing in virtual organizations. It has developed algorithms and software for building and managing semantic link networks. It developed a single semantic image technique based on normalization theories to support service integration. The project team also proposed the *active document framework* (ADF) to enable active information services.

3.3. *GridDaen*

Grid Data Engine (called GridDaen) [23], which is implemented by National University of Defense Technology (NUDT), is a data Grid middleware which can integrate various kinds of file systems and provides uniform seamless access to distributed datasets. It implements logical name spaces for users, resources, files, and collection, and provides data replica and caching mechanism to improve data access performance in distributed heterogeneous environment. GridDaen supports uniform and secure access and management of various types of heterogeneous distributed storage and data resources, such as file systems (e.g., Linux ext2, Windows NTFS), network file

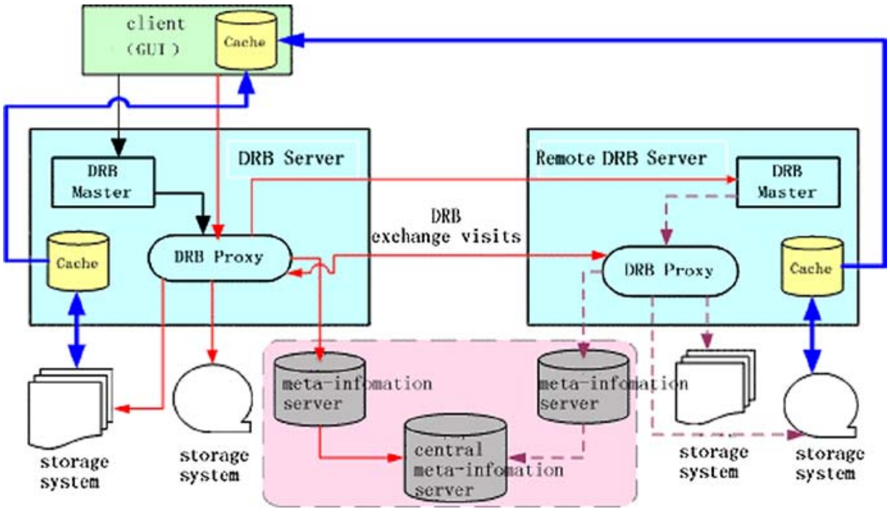

Figure 3. The components of GridDaen.

systems (e.g., NFS). Also it provides global views and convenient, standard access and management APIs and interfaces for users and applications. Federated data servers and distributed metadata information servers across multiple domains, are implemented by GridDaen to support autonomy of distributed dataset resources. The system is being deployed in China National Grid and will be extended to support database systems in 2004.

The main components of the GridDaen system include Client tools, DRB (Data Request Broker) servers and metadata information servers (MDIS), as illustrated in Figure 3. It is implemented in Java.

The Client tools of GridDaen provide a set of API packages, command lines and GUI tools for users to develop applications and manage data.

The DRB data servers mainly provide data access and management of the heterogeneous distributed storage and data resources through uniform data access interface, high-speed data transfer, replica management mechanism and virtual datasets, etc. There is a DRB server to independently act to perform data operations on local data resources in each administrative domain. The DRB server is composed of two components: DRB master and DRB proxy. DRB master receives clients requests and maintains the global information of the DRB. When a request from clients arrives, the DRB master receives the request and validates the security characteristic of the request, and then assigns a DRB proxy to serve the request. The DRB proxy performs the actual data operation according to the request, such as data access, transfer and replication. Multiple DRB servers can collaborate to form a federated DRB which acts like one single server and provides uniform services for data requests. User's request can be sent to any DRB server. The DRB server analyzes users' request, then sends it to the corresponding DRB server that is responsible for the resources the request requires. Once the DRB has acquired data, it directly transfers data to client by high-speed data transfer protocol. GridDaen adopts GSI to support single sign-on authentication.

The MDIS are distributed metadata servers to provide query and maintenance for whole data Grid system resources information, user metadata, security and authorization information, replica information, etc. It is organized in a distributed structure which composes of local metadata servers and central metadata servers. Each local metadata server can independently provide the metadata service of local resources. The central metadata server only builds the indices and maintains caches for local metadata servers to achieve global view and location-transparent access. Users can query resource information from any local metadata server in the system rather than from the central metadata server. When the required information could not be found in the local metadata server, the query is automatically redirected to the central server. Some GUI tools are provided for users to publish resource and data, authorize, configure and deploy system.

3.4. *HowU*

HowU network computing platform is a global computing paradigm based on volunteer computing model. It is developed by Cluster and Grid Computing Lab at

Huazhong University of Science and Technology, aiming to solve the problem of existing network computing platforms, such as SETI@Home [17], GIMPS [6], D2OL [3], Folding@Home [5], and XtremWeb [4].

Adoption the mechanism of the application system separated from core software by the server system and the volunteer system makes applications make use of computing resources of the platform for computing. The characteristics of server system include high availability, flexibility and low price. The self-adaptive fault-tolerant backup system [18] based on Linux Virtual Server (LVS) [12] improves fault tolerance feature of the system greatly. The volunteer system which is built on top of Windows and Linux makes most computers over Internet become potential volunteers of the platform. Friendly designed the volunteer interface system makes volunteers control volunteer software conveniently. Computing thread starting up with lowest priority in background would not disturb the normal working process of volunteers.

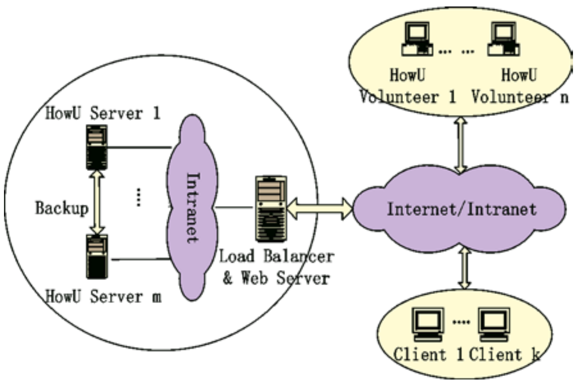

Figure 4. Architecture of HowU Network computing platform.

The architecture of HowU platform is shown in Figure 4. The platform is composed of three parts: the server system, the volunteer system, and clients.

Being an open network computing platform, clients could submit applications to the server system according to specific format. The server system is built with improved LVS cluster architecture. One of the cluster nodes works as a load balancer, others works as HowU servers. All volunteer requests are sent to the load balancer first. According to the load balancing policy, the load balancer relays the requests to a right HowU server. After that, the HowU server will send the client request to the volunteer node in the HowU system directly. Cluster architecture of the server system is transparent to volunteers.

The volunteer working system is composed of three function modules: control module, communication module, and computing module, shown in Figure 5.

The control module is responsible for running of the volunteer software. It initializes and starts up the communication module; creates the working pool, and manages static tasks. When the condition of starting up computing is met, the control module starts up the computing process at once. While the condition of starting up computing is not met, the control module stops the computing process immediately.

The communication module is responsible for communication with the server. It connects with the server, and executes communication events one by one from the communication queue. Communication events include registration of the volunteer, information modification of volunteers, registration of the volunteer machine, asking for tasks, sending alive message, and returning results.

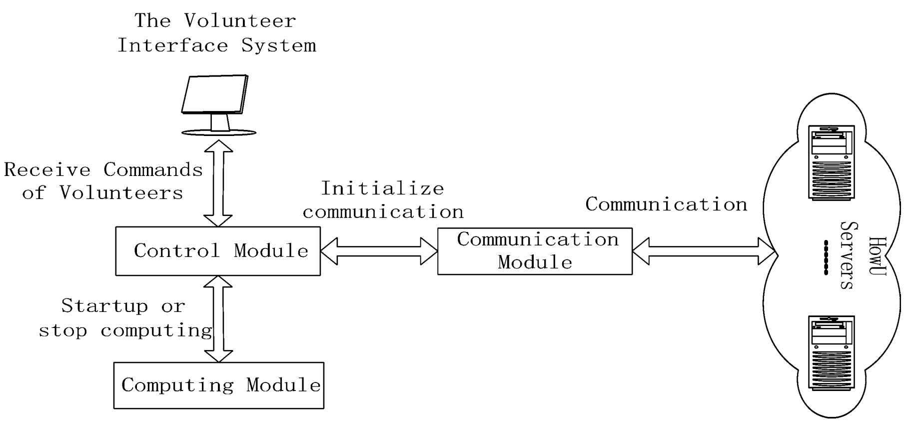

Figure 5. Function of the volunteer working system.

The computing module is responsible for starting up and stopping of the computing thread. When the computing thread is started up, it fetches a task from the task pool, creates a temporary working dir, then starts up computing, and writes the result to a designated file. After a task is fulfilled, the corresponding computing thread dies. The computing module will get next task from the task pool.

A computing thread is created with the lowest priority. On the one hand, it harvests idle cycles of the system fully; on the other hand, it will not affect the working of volunteers. Once any work of the volunteer needs computing, the computing thread is paused at once.

In our experiments, we use the DES decryption application running on HowU platform. According to the range of data in character directory, we divide the DES decryption program into 19200 independent subtasks. Every subtask executes 4218750 DES decryption operations. The computing amount for each subtask is equal to a PC with 450 MHz running 10 minutes in full speed. Making use of HowU platform, we distribute the subtasks to volunteer machines over Internet, and compute by harvesting idle cycles from them. In two days experiments, HowU servers receives 179 volunteer registrations, fulfills 7669 subtasks. The total computing power is equal to a PC with 450 MHz running 53 days in full speed.

3.5. *DartGrid*

DartGrid [9], built upon several semantic web standards and Grid technologies, is aimed to provide a semantic-heterogeneity solution that addresses the challenge of web-scale cross-enterprise database integration. DartGrid, developed by Grid Lab at Zhejiang University, has the following features: (a) database providers are organized as an ontology-based virtual organization; by uniformly defined domain semantics, database could be semantically registered and seamlessly integrated together to provide database service, and (b) we raise the level of interaction with the data base system to a domain-cognizant model in which query request are specified in the terminology and knowledge of the domain(s), which enable the users to publish, discovery, query databases only at a semantic or knowledge level, (c) GSI is directly integrated with traditional DBMS security mechanism to deal with the cross-enterprise authentication and authorization.

DartGrid is mainly constituted by the following components:

- *Data Semantic Service (DSemS). Data Semantic* could be viewed as local schema, which is generated by transforming relational schema into corresponding RDF/OWL semantic. Local DBA publishes local data semantic for a specific database by DSemS. Others could inquire of it to fulfill some tasks such as browsing, inquiring or semantic mapping.
- *Ontology Service (OntoS).* Ontologies could be viewed as mediated schema, and contain the widely-agreed salient aspect of the domain under consideration. Ontologies are published by OntoS for user to browse or inquire.
- *Semantic Registration Service (SemRS).* Semantic registration establishes the mappings from local data semantic to mediated ontologies, from ontologies to other ontologies (hierarchical mode), or from local data semantic to other target data semantic (P2P mode).
- *Semantic Query Service (SemQS).* Semantic Query is a query specified in terms of ontologies or local data semantic, i.e., a query in term of RDF/OWL semantic. Typically, SemQS accepts a semantic query to SemRS to determine which lower-level SemQS is capable of providing the answer, then rewrites the query in terms of lower-level ontologies and generates a series of sub semantic queries, which are then dispatched to those SemQS involved. Finally, the queries will be delivered to the SemQS of a specific database where the queries will be rewritten using relational tables or views.
- *Semantic Browser (SemB).* DartGrid also provides user with a semantic interface to graphically browse RDF semantic, and visually construct a semantic query.
- *Query3 (Q3).* Q3, named after Notation3 (N3), is a semantic query language. Typically, user uses SemB to visually construct a Q3 query and submit it to SemQS for query processing. Programmers could also manually write a Q3 query and embed it directly in their programs.

3.6. *Comparison*

In this section, we will make a comparison between the programs listed above.

T.G and HowU are common in that they are both closely related to computational Grid. Resources executing real computation in these two systems are usually common PCs with no existing local resource management policy. The applications suitable to run in these two systems are those with divisible loads, i.e. the whole application is composed of several independent subtasks. The difference lies in that T.G is much more distributed than HowU. In Grid constructed with T.G, there is no centralized server. A user can submit an application to any node in the system, which will then collaborate with other nodes to complete the request. Whereas in HowU, a centralized cluster server is responsible for accepting user's request and direct it to appropriate resources.

One of the characteristics of Vega lies in that it makes Grid-oriented research in hardware and builds Dawning 4000A, a Grid-enabled cluster server. Besides, Vega is also making attempts in knowledge Grid to knowledge management and decision support.

GridDaen is targeted for data Grid, with the aim to integrate various kinds of file systems and provide uniform seamless access to distributed datasets.

4. Grid Applications

The ultimate goal of Grid computing is to effectively support its applications and provide key productive applications. To evaluate the practicability and demonstrate the capability of key programs presented in above section, and more importantly, to resolve practical problems in research, industry and municipal infrastructure, many industrial applications have been developed on top of them. Due to space limitation, we present four representative applications in this section.

4.1. *Bioinformatics Grid*

Bioinformatics is a cross discipline combining biology, computer science and information technology. It usually processes gigantic heterogeneous data through massive computation, using some common tools. Unfortunately, each bioinformatics research institute usually sets up and uses such environment for its own use, which leads to a low utilization of them. However, many researchers in universities in China even have no such research resources to use.

The Bioinformatics Grid developed by Tsinghua University focuses on existing valuable large-scale heterogeneous computers, storage facilities, data and common used software tools distributed across 12 top China universities, so as to make them available where and when they are needed, and make them cooperate transparently, resulting in a virtual miscellaneous bioinformatics service center to all licensed users in ChinaGrid.

The Bioinformatics Grid provides a high performance computing platform, which makes some bioinformatics software tools in common use, for example BLAST and PHRAP, network based, and provides some basic computation services such as genetic forecast, makeup analysis, sequence comparison, repetition sequences searching, control zone prediction and function classifying in bioinformatics transparently.

Moreover, this project also aims to construct a Grid-based biological computation environment. In such an environment, distributed, heterogeneous computer nodes, as long as they install certain biological computation programs and are willing to share them for public use, can be aggregated seamlessly into a computational Grid to provide biological computation services. It provides a general and practical application model. Just with the widely available web browser, biology researchers can conveniently utilize both powerful software and hardware resources contributed by a collection of diverse service providers.

The project also solves the problems such as user management, resource distribution, task control and results feedback under multi-user and multi-task environment.

Above all, the Bioinformatics Grid holds the following features.

- Accessing to bioinformatics computation service through Internet.
- Improving the utilization and service quality of the facilities in the Grid.
- Sharing abilities of bioinformatics software in different OSes.
- Providing cooperative computing service between bioinformatics tools through a unified interface and a workflow engine.

Current applications of Bioinformatics Grid include blast of unknown sequences in Cordyceps sinensis genome, pairwise alignment of HHV1 and HHV2 sequences of herpesvirus, sequence splicing of rat genome, Grid computation for gene identification and so on.

4.2. *Scientific Data Grid*

Scientific Data Grid (SDG) is an application developed by Network Center of Chinese Academy of Science, whose goal is to realize effective sharing of heterogeneous scientific databases distributed across more than 40 institutes subjected to Chinese Academy of Sciences, and to develop some application systems that have practical importance for scientific research. The main achievement of SDG lies in the following aspects:

Construct a system platform for Scientific Data Grid. This platform is the fundamental of the Scientific Data Grid, which comprises over 10 TB scientific data, 15 TB network storage capacity, a data center node and three data sub-center nodes.

Develop middleware of Scientific Data Grid. The middleware settles the problem of sharing scientific data resources in Grid environments, which is composed of Information Services System, Uniform Access Interface and Security System.

Information Services System is the core of the scientific data Grid middleware. Its primary function is to provide unified information services for data resources in scientific data Grid, especially metadata information service. Information Services System is compatible with and extended from MDS, which is a part of the Globus Toolkit and has been widely deployed and used.

The function of Uniform Access Interface is to implement a uniform access interface for distributed and heterogeneous resources. Uniform Access Interface relies on the above Information Services.

Security System is the premise of resource sharing. A complete, credible, convenient and efficient security system is crucial to a Grid, in particular, the Scientific Data Grid. Security System of Scientific Data Grid middleware is developed based on GSI (Grid Security Infrastructure).

Above all, the middleware provides a uniform access interface for distributed and heterogeneous scientific databases, including Oracle, Microsoft SQL Server, file systems and multimedia databases. It also offers unified information services for data resources in scientific data Grid.

Develop a demonstration application system of Scientific Data Grid, i.e. China Virtual Observatory (CVO). The goal of CVO is to integrate and utilize the national major astronomical research resources transparently, taking advantages of Grid technology and services provided by Scientific Data Grid middleware.

CVO realizes seamless federation and integration of multi-band astronomical observation archives, and makes deep-level analysis and process with shared data resources. At the same time, CVO will provide some tools and services for spectrum auto-processing and data mining.

4.3. *Traffic Congestion Control*

For the purpose of integrating resources in current major computational aggregations and networks in Shanghai and forming a uniform information Grid to provide metropolis applications, especially the traffic-congestion control and guidance application, the ShanghaiGrid project needs to make variety of heterogeneous devices (supercomputers, workstation and desktops, PDAs, sensors and data collectors, etc.) and diverse virtual organizations (Intranet, campus networks and database, etc.) intercommunicate and collaborate into an efficient environment. To achieve the goal, the project implements OGSA-based protocols to form several Intra-Grids with powerful computers as the central nodes and other devices, as shown in Figure 6. These Intra-Grids play their coresponding roles in ShanghaiGrid, such as data collection, data storage and computation supply. On top of these localized Intra-Grids, ShanghaiGrid constructs a uniform information Grid environment to provide global information services and upper-layer applications, as shown in Figure 7.

Based on the global information services in ShanghaiGrid, a specific Grid-based application in the realm of traffic management [11, 19, 21, 16, 20] is implemented, i.e., the application for traffic-congestion control and guidance. This is the most important application in ShanghaiGrid, and also a valuable project for

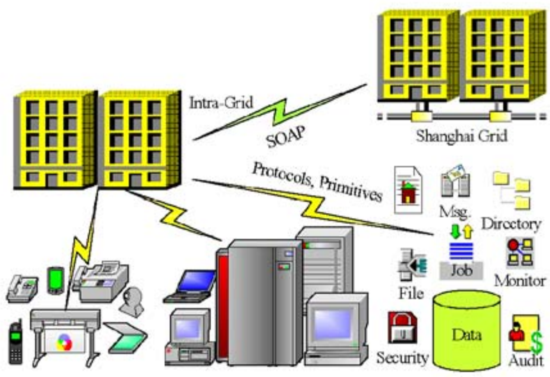

Figure 6. Several Intra-Grids in ShanghaiGrid. Each Intra-Grid comprises central supercomputers or powerful workstations as the central nodes (e.g., SW-I, SUHPCS, IBM P690, IBM E1350, SGI Origin 3800), other devices (e.g., GPS sensors, traffic surveillant, mobile phones), software packages (e.g., PBS, LFS) and other resources. Intra-Grid abstracts different resources as virtual services.

Shanghai government. The traffic-congestion control and guidance application have the following functionalities:

- Traffic data are distributed, dynamic, and of great volume. In this project, all these data should be collected, transferred, stored, aggregated by all kinds of resources and services within ShanghaiGrid.
- Model the traffic status and traffic-jam accurately, dynamically, adaptively and in time.
- Analyze and forecast traffic status from the massive information data.
- Simulate traffic status accurately for the decision-making of the government.
- Provide traffic information on demand services and publish traffic status to community, vehicles and passengers.

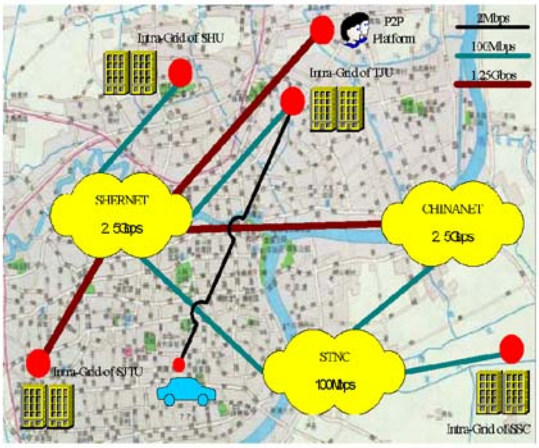

Figure 7. ShanghaiGrid connects several major Intra-Girds to form a sophisticated information environment, which includes Shanghai Jiao Tong University, Tongji University, Shanghai University and Shanghai Supercomputer Center. Other sites could be connected to some of the major Intra-Grids in ShanghaiGrid.

In the application, traffic information could be published and subscribed in various ways, such as using mobile phone, through broadcasting program, by means of live bulletin board. Typical GUIs of the application of traffic-jam guidance and control are shown in Figure 8.

4.4. *Traditional Chinese Medicine Grid*

Chinese medicine, a complete medical system that has diagnosed, treated, and prevented illness for over twenty-three centuries, has provided us a wide variety of resources for biomedical and health science. These resources include TCM literatures, medicinal materials, Chinese herbs, TCM compounding rule, Chinese medical formula and so on, both ancient and present. In China, there have been thousands of TCM hospitals, enterprises and research institutes who have developed various TCM databases and other information product to serve people's health care and research demands.

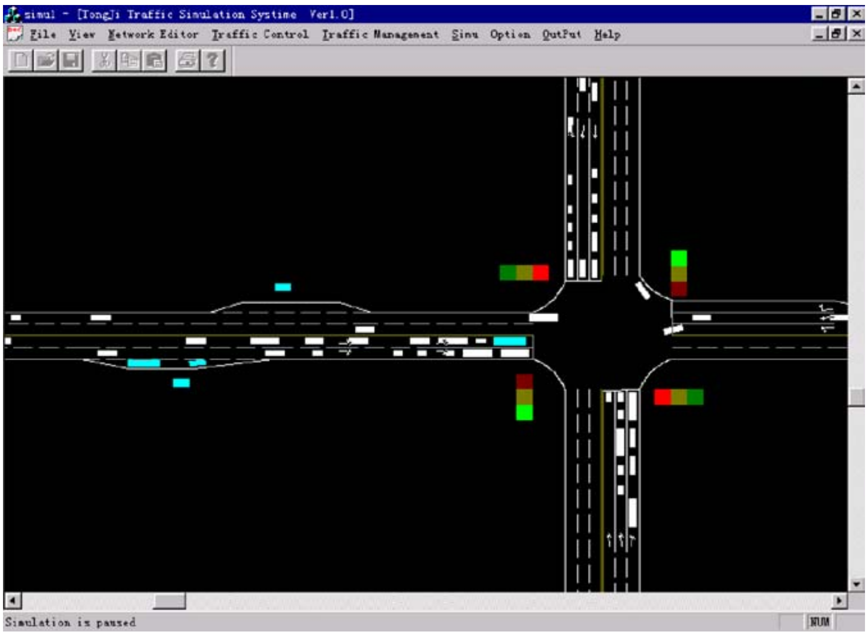

Figure 8. This GUI simulates the traffic status in the crossroad. Road surface, traffic lights, and vehicles flows are shown in the scene.

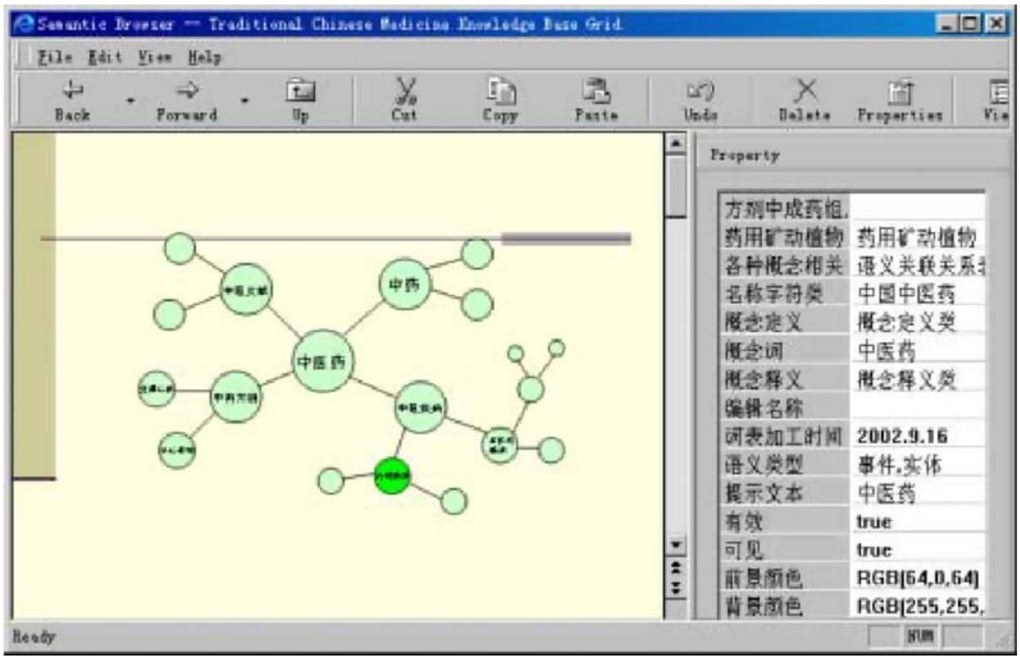

Figure 9.

TCM-Grid [22] aims to aid the development of distributed systems that help health professionals, researchers, enterprises and personal users to retrieve, integrate and share TCM information and knowledge from geographically decentralized TCM database resources and knowledge base resources in China. TCM-Grid is used to facilitate knowledge sharing and database resource integration within TCM communities. TCM-Grid consists of the following basic elements.

- TCM ontology Service: TCM ontology service is a public available reference service. The TCM ontology provides service of querying TCM terminology, for examples, synonymy or subsumption relationship between TCM concepts.
- TCM KBs and DBs: TCM KBs and DBs store the knowledge and data about Disease, Medicine, Prescription, Literature, etc., respectively. They include Symptom KB, Therapeutic Principle KB, Medical Formula KB, Compound Rule KB, Therapeutic KB and so on in the TCM domain.
- TCM Semantic Registration Service: TCM Semantic Registration Service provides a standard mechanism for registration and discovery of the KBs and DBs in TCM community. All the TCM KB and/or DB, before participating in the community, should register themselves to this service.

5. Conclusion

In recent years, Grid computing has become a rapid growing research area. We need Grid for greater computing capacity, resource sharing, collaborative work, utilizing idle computing power, etc. China views Grid as an important opportunity for science, education, industry, and marketing. Grid computing is a very active field in China, where both national Grid plans and industrial/campus research projects have been launched to build Grid systems, covering a wide range of Grid techniques, especially the middleware and practical applications.

References

1. "ChinaGrid", `http://www.chinagrid.edu.cn/`
2. "China National Grid", `http://www.grid.org.cn/863_105/applygide/applygide2/information2_app/200206270024.html`
3. "Drug Design and Optimization Lab (D2OL)", `http://www.d2ol.com/`
4. G. Fedak, C. Germain, V. Néri and F. Cappello, "XtremWeb: A Generic Global Computing System", in *Proceedings of Workshop on Global Computing on Personal Devices*, Berlin, Germany, 2001, pp. 582–587.
5. "Folding@Home Project", `http://www.stanford.edu/group/pandegroup/folding/`
6. "GIMPS Project", `http://www.mersenne.org/prime.htm`

7. F.S. Graham, C. Kesselman, T. Maguire, T. Sandholm, D. Snelling and P. Vanderbilt, "Open Grid Services Infrastructure", OGSI V.1 Specification: GFD.15.
8. Y. Han et al., "CAFISE: An Approach to Enabling Adaptive Service Configuration of Service Grid Applications", *Journal of Computer Science and Technology*, Vol. 18, No. 4, 2003, pp. 484–494.
9. C. Huang, Z. Wu et al., "Dart: A Framework for Database Resource Access and Discovery", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3033, Shanghai, December 2003, pp. 855–862.
10. M. Li, H. Liu, C. Jiang et al., "ShanghaiGrid in Action: the First Stage Projects Towards Digital City and City Grid", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3032, Shanghai, December 2003, pp. 616–623.
11. M. Li, F. Tang et al., "An Integration Architecture for Grid Resources", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3032, Shanghai, December 2003, pp. 708–715.
12. "Linux Virtual Server Project", `http://www.LinuxVirtualServer.org/`
13. "NFSC Grid", `http://www.nsfcnet.net/`
14. "OGSA Working Group in GGF", `http://forge.gridforum.org/projects/ogsa-wg/`
15. "OGSI Working Group in GGF", `https://forge.gridforum.org/projects/ogsi-wg/`
16. R. Rao, B. Li, M. Li and J. You, "The Delivery and Accounting Middleware in the ShanghaiGrid", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3032, Shanghai, December 2003, pp. 786–793.
17. "SETI@Home Project", `http://setiathome.ssl.berkeley.edu`
18. G. Tan, H. Jin and L. Pang, "Layer 4 Fault Tolerance: Reliability Techniques for Cluster System in Internet Services", in *Advanced Environments, Tools, and Applications for Cluster Computing*, Lecture Notes in Comput. Sci. 2326, Springer, pp. 61–69, March 2002.
19. F. Tang, M. Li and J. Cao, "A Transaction Model for Grid Computing", in *Proceedings of the Fifth International Workshop on Advanced Parallel Processing Technologies*, Lecture Notes in Comput. Sci. 2834, September 2003, pp. 382–386.
20. F. Tang, M. Li and J. Cao, "A Transaction Coordination Mechanism and its Compensation Technology in Grid Environment", *Journal of Computer Research and Development*, December 2003 (in Chinese).
21. F. Tang, M. Li et al., "Coordinating Business Transactions for Grid Service", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3032, Shanghai, December 2003, pp. 108–114.
22. Z. Wu et al., "Knowledge Base Grid: A Generic Grid Architecture for Semantic Web", *Journal of Computer Science and Technology*, 2003.7.
23. N. Xiao, D. Li, W. Fu, B. Huang and X. Lu, "GridDaen: A Data Grid Engine", in *Proceedings of the Second International Workshop on Grid and Cooperative Computing*, Lecture Notes in Comput. Sci. 3032, Shanghai, December 2003, pp. 519–528.
24. Z. Xu et al., "Dawning Superservers and the Vega Grid", Keynote speech paper, in *Proceedings of HPC-Asia 2002*, Bangalore, India, December 2002, pp. 1–7.
25. H. Zhuge, "Semantic, Knowledge and Grid" (Editorial of the special issue on Semantic Grid and Knowledge Grid: the Next-Generation Web), *Future Generation Computer Systems*, Vol. 20, No. 1, pp. 1–6, 2004.