A STUDY ON ENTERPRISE APPLICATION INTEGRATION BASED ON GRID

Dan Chang and Lijun Zhou
Institute of Information Systems, Beijing Jiaotong University, Beijing, China

Key words: Grid Technology, Enterprise Application Integration, EAI-Based-Grid.

Abstract: Enterprise Application Integration (EAI) is a kind of method and technique which can solve the problem of information sharing. However, the existing EAI solutions generally have some limitations. For example, when the enterprises need to implement cross-boundary information integration between different systems, they cannot meet the needs. The grid technology can achieve cross-boundary, cross-platform and distributed resource sharing, also provide complete security mechanism. So the enterprise application integration based on grid technology is able to solve the current problems. In this paper, some existing studies on EAI solutions will be introduced. And then grid technology will be applied to the enterprise application integration to form a solution EAI-Based-Grid, at last we analyze the characteristic of this method and conduct some simulations to prove its feasibility.

1 INTRODUCTION

With the rapid development of e-commerce, enterprises have adopted ERP, CRM, PDM and other information systems successively which greatly improve the level of information technology application. However, these systems may be implemented by different companies and only focus on their respective data and business processes. Because of the lack of corresponding standard specifications for interface, they cannot be carried out information sharing and business integrations each other. Therefore, companies need an increasing strong and interconnecting IT infrastructure to face to new opportunities and challenges. Enterprise Application Integration (Qureshi, 2005) (EAI) is a kind of method and technique which solves the matter of information exchange and sharing. It uses an open, extensible system to integrate enterprise software and hardware, finally achieve the resource sharing and business processes.

The present EAI solutions (Hub-And-Spoke, message bus, web services (Wu and Yang, 2008), etc.) may partially meet the enterprises' demands for information sharing between systems, while there are limitations of the application, such as the platform restrictions, geographical restrictions or safety visits. Especially when companies need to achieve cross-boundary information integration between different systems, this limitation is particularly prominent. However, the grid technology can implement cross-boundary, cross-platform and distributed resource sharing, also provide complete security mechanism (Zhang and Bao, 2008). The enterprise application integration based on grid technology can solve the current problems.

In this paper, we propose the advantages and disadvantages of existing EAI solution. Directing at their limitations and combining with the advantages of grid technology, we propose enterprise application integration solution based on grid through absorbing the essence of Reference 4. And then we conduct simulation about this method. The simulation results prove the method that grid technology is applied to enterprise application integration is feasible.

The second part in this paper briefly describes the existing studies of enterprise applications integration. The third part explains the theories and knowledge about grid technology. According to these two parts, enterprise application integration based on grid is proposed in the forth part. Then in order to prove the feasibility of this method, a simulation is carried out to demonstrate it. At the end of this paper, it is a general summary.

Zhou L. and Chang D.
A STUDY ON ENTERPRISE APPLICATION INTEGRATION BASED ON GRID.
DOI: 10.5220/0003268802920298
In *Proceedings of the Twelfth International Conference on Informatics and Semiotics in Organisations (ICISO 2010)*, page
ISBN: 978-989-8425-26-3

2 EXISTING STUDIES ON EAI SOLUTIONS

The existing EAI solutions include Hub-And-Spoke, message bus, web services, etc. All of them have their own advantages and disadvantages.

2.1 Hub-and-Spoke

In the Hub-And-Spoke structure, different application programs are connected to a central server. As is shown in Figure 1 (Gu et al., 2004), a variety of relational databases, customer relationship management systems and so on are connected to a central server through the adapter. The server is like a hub, it is similar as the star structure of network topology. Central server plays the role of a message broker and it is mainly responsible for the communication management, data transmission and process between multiple systems. A new system or application program can be connected to other hub through an adapter.

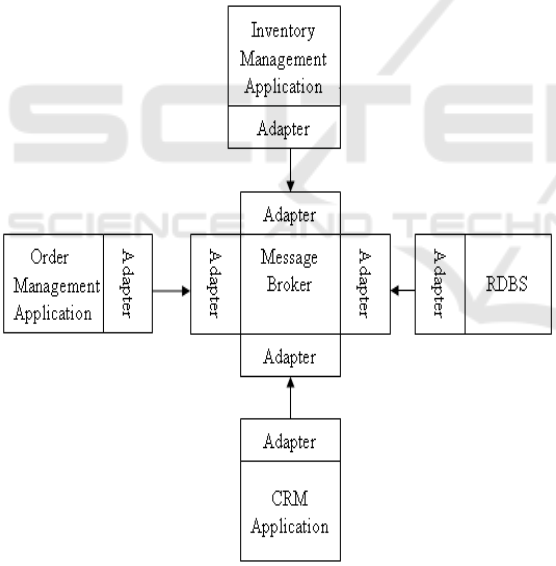

Figure 1: Hub-And-Spoke structure.

Hub-And-Spoke structure possesses some advantages such as centralized control, ease of administration and so on. However the disadvantage is that the server easily becomes a bottleneck in the system. The whole system's stability will be gradually decreased when the new system is connected.

2.2 Message Bus Structure

In the message bus structure, all systems are connected to the virtual message bus via adapter. The adapter will convert the internal events of application program into message and then exchange the message with other application programs via message bus. Meanwhile it will receive the message and convert the message into internal events. It is shown in Figure 2 (Gu et al., 2004).

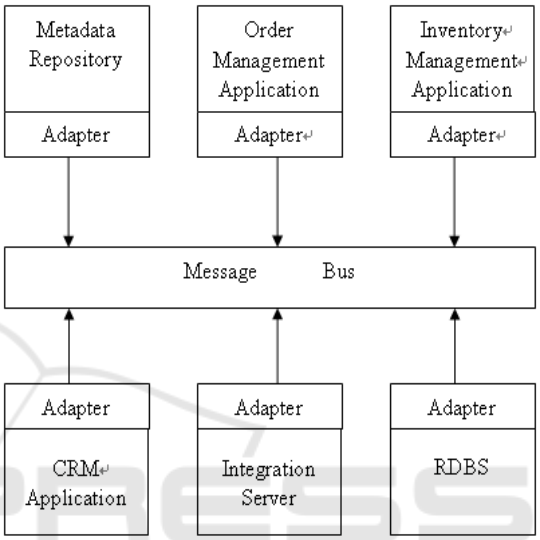

Figure 2: Message bus structure.

The advantage of message bus structure is that it has no communication bottleneck, the adding of new system is easy to do and scalability is wonderful. On the contrary the drawback is that the implementation is complex.

2.3 Web Service

The web services structure can integrate enterprise applications (Wu and Yang, 2008). The discovery, description and transmission of services are run by web services based on UDDI, WSDL and SOAP agreement of XML. This system reduces the coupling of enterprise application integration. It is shown in Figure 3.

Web services structure has the advantage of low coupling, good cross-platform and a wide range of application. However, security, transaction and workflow are its weak point.

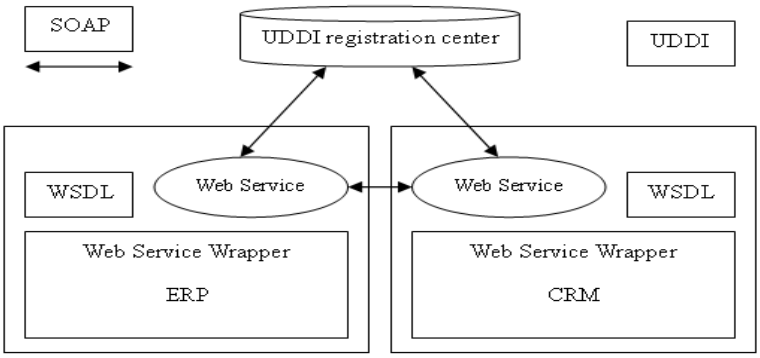

Figure 3: Web service structure.

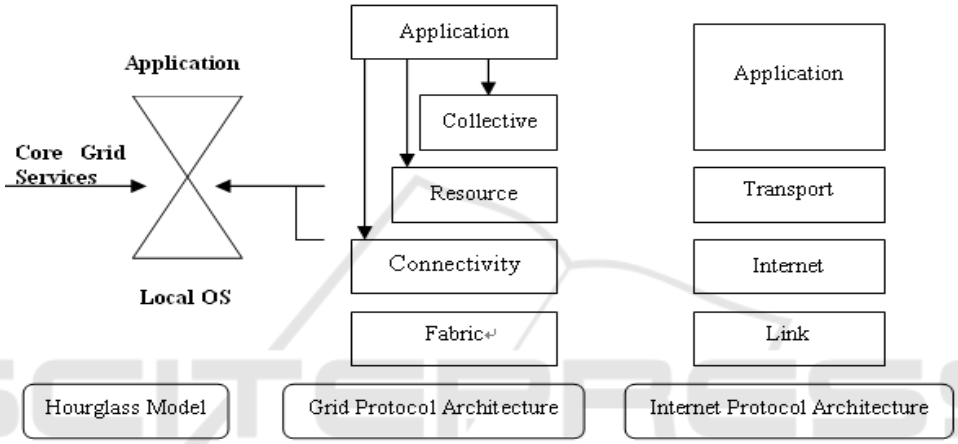

Figure 4: Hourglass model.

3 RELATED THEORIES ON GRID

3.1 Grid

Grid is originally used for large-scale scientific and engineering computing, and which now has developed into an important technology for distributed resource sharing. The integration of remote computing resources can be carried out by grid technology. Then the dynamic resources sharing and problem processing of multi-agency virtual organization will be realized 。 The problem mentioned here may involve data processing, data storage or business processing (Iosup et al., 2006).

A system can be named grid must meet the following three points (Foster et al., 2001): i) Sharing the non-centralized and synergic resource. ii) Using standard and open protocols and interfaces. iii) Providing wonderful quality of service.

3.2 Standard and Architecture of Grid

The international organization which draws up grid computing standards, Global Grid Forum, has released standard system called OGSA (Benson et al., 2006) (Open Grid Service Architecture) and OGSI (Wang et al., 2004) (Open Grid Service Interface), which provides the structure and interface of grid respectively.

The major problem of distributed resource sharing is interoperability. In the network environment, interoperability depends on the common agreement. Therefore, the grid architecture involves the structure of related agreement first, and then the working mechanism is built on the agreement structure. The design of grid architecture follows the Hourglass Model principles (Zhang and Bao, 2008). It is shown in Figure 4.

In the grid environment, there are a large number of temporary services, such as a computing task execution. Considering the specific characteristics of grid environment, OGSA propose grid service

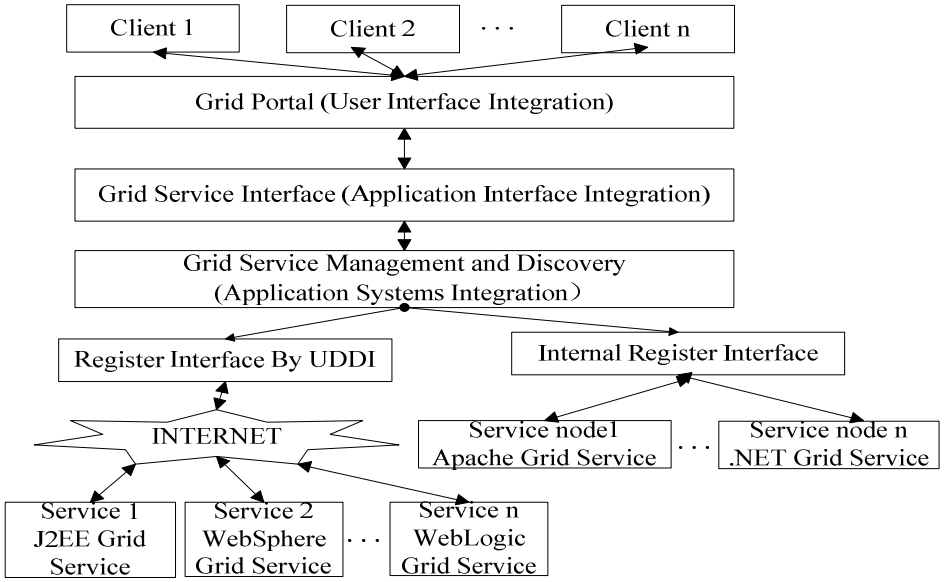

Figure 5: Framework of EAI-Based-Grid.

based on original web services. It is used to solve the issues which are related with temporary service such as service discovery, dynamic service creation, and service life-cycle management.

4 EAI BASED ON GRID TECHNOLOGY

4.1 Structure of EAI-based-Grid

Considering the demerits of the previous three kinds of EAI structure and the merits of grid, EAI based on grid technology can be proposed. EAI-based-Grid adopts OGSA as basic framework. In OGSA, everything, including computers, programs, data and equipment so on, is abstracted as service. Globus Toolkit software package and Web Service are two supporting technologies for the construction of OGSA. Globus Toolkit is a set of services and software library that supports grid computing and grid application. Web Service is commonly used as a standard framework to access network applications. The framework of EAI-Based-Grid is shown in Figure 5.

After entering grid portal through web browser, the client can see the unified data view of system, access and manipulate the data, and call the grid service provided by system. The communication between the client and the grid server is carried out by a grid service interface. The transfer protocol between them is the SOAP protocol and the data is described by XML language.

In this structure, programming language of grid service interface can use the high-level service programming language provided by Globus Toolkit. This interface offers methods of the access to all internal grid resources. Every service node has an operation platform of service, such as Apache, J2EE, WebSphere and so on, that supports Grid Service and conducts interaction with the upper layer (Grid Service Management and Discovery layer).

4.2 Characteristic of EAI-based-Grid

(i) Grid portal simplifies the process of access to resources. It is convenient for user to interact with multiple systems.

The client can find all the resources in the system through the grid portal, including grid services, operational status, technical documentation, etc. The users who have a system account and be authorized can add and manipulate their files and data, also can submit, monitor and abolish the interaction with the relevant application system. In this way, users can achieve seamless access to system resources through web browser. However, the other enterprise application integration is difficult to achieve this result.

(ii) The application interface layer is intergraded through grid service interface, which improve the intelligent level of the application system.

When a user requests grid services through a grid portal, grid interface convert the user's request to

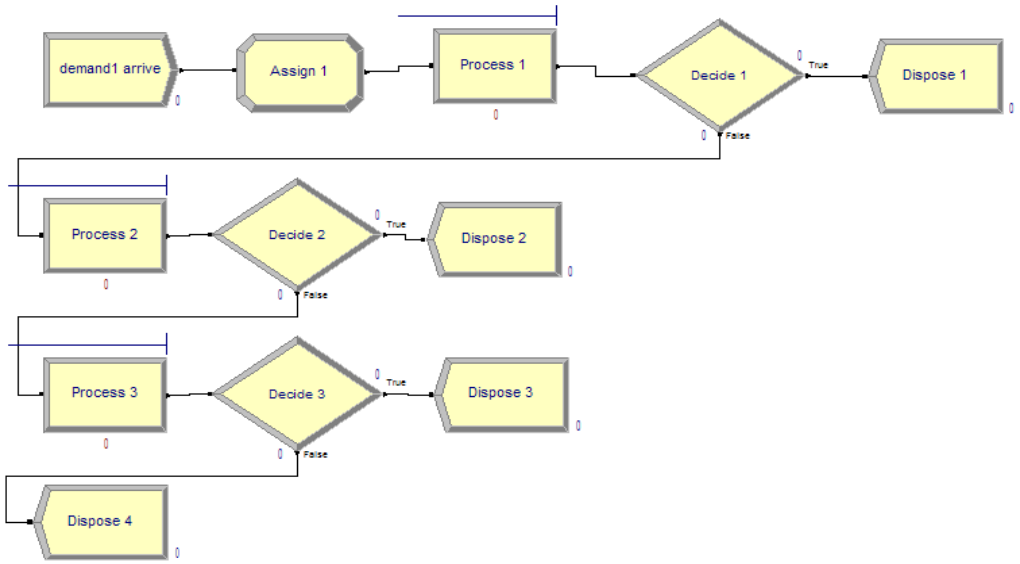

Figure 6: Process of simulation model one.

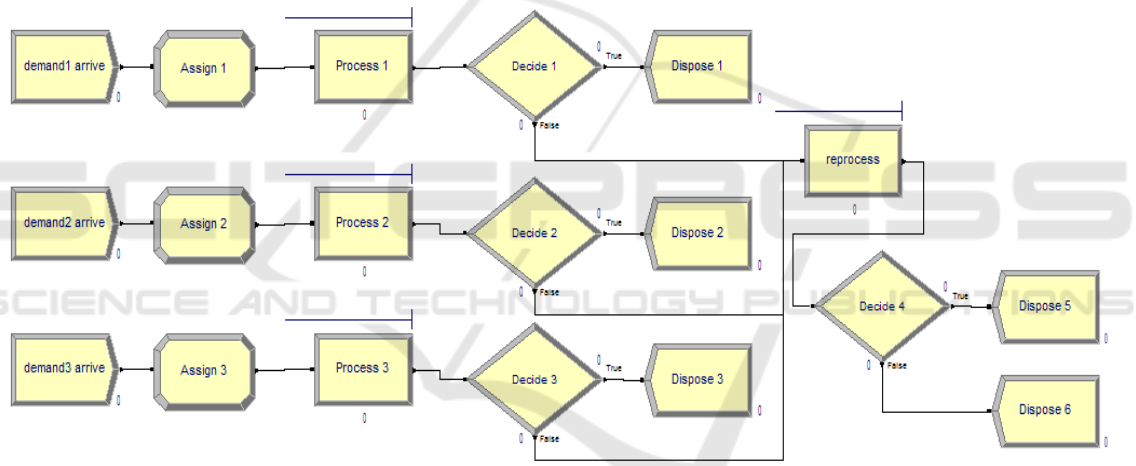

Figure 7: Process of simulation model two.

XML files, send it to the task analysis module and then form a macro-task workflow. For every stage of the workflow, analysis module will query whether corresponding grid services are included through the grid service management and discovery layer. If these services are found, the task will be optimized first and then a grid services list related to the task will be made. Next these services will be called and the results of the service will be saved in Cache. Finally it will be submit to the client through the grid portal. If these services are not found, an error returns. In this way, the dynamic integration of application interfaces is realized and the intelligent level of the system is improved.

(iii) Grid service management and discovery layer uses the popular UDDI technical specifications, which is convenient for integration with existing systems.

It takes use of service registry technology and UDDI standard to complete grid service registration of each server node. If the grid needs to perform specific calculation, the grid service management and discovery layer will register the service to node that owns such service and activate the service.

(iiii) Server nodes based on grid can provide a variety of grid services and improve comprehensive utilization of various resources in the system.

These services include hardware status services, message services, basic business function library services and application services. These services can

be aware by grid service management and discovery layer after they are registered, then be managed and distributed by grid service interface layer and finally be provided to clients by grid portal. Clients can take use of all the resources of the system in accordance with their permissions. It improves the rate of multipurpose utilization dramatically.

5 SIMULATION AND ANALYSIS

To verify the advantages of EAI-Based-Grid mentioned above, we need make the solution analog abstracted and then we can carry out simulation and analysis on it. And in order that simulations between based on non-grid and grid have comparability, we assume that the resources and the parameters in the two systems (non-grid and grid) are the same.

5.1 Simulation Model

Model One: Simulation model based on Non-grid. In this model, if the user requests service X, he need to query n (i=1, 2… n) resources of each server in order to find the node that can meet the demand. Resource i has m links such as occupation, delay, release and so on. All resources subject to the FIFO rule. If all resources are occupied, it will form a queue. The client can not get the service until the resource is idle. The process of simulation model is shown as Figure 6.

Model Two: Simulation model based on Grid. In accordance with the grid management theory, this model integrates business process to form a unified control center. Then enterprise business integration and resource sharing will be realized.

In this model, if the user requests service X, the service can be aware by grid service management and discovery layer after register. Then the service is managed and distributed by grid service interface layer. Finally, the service is provided to client through the grid portal. If the resource X is occupied, a single queue will be formed. If the resource X is idle, the system will arrange it for the user. The process of simulation model is shown as Figure 7.

5.2 Simulation Results

Based on the above model, the results of simulation using Arena software are shown as Figure 8, Figure 9, and Figure 10. The horizontal axis represents system resources, the vertical axis respectively shows three different indexes, Average Length of Queue, Average Sojourn Time, Level of Resources Utilization.

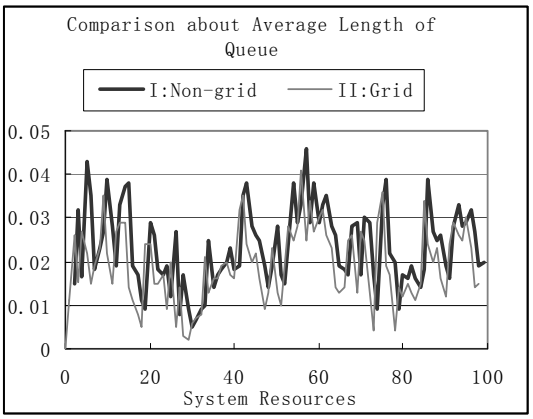

Figure 8: Comparison about Average Length of Queue.

The Average Length of Queue L: $L = \lim_{T \to \infty} \int_0^T \frac{(_{Lq}(t) + S(t))\,dt}{T}$ [S (t) represents customers who are receiving service. $_{Lq}(t)$ represents customers who are waiting]. The simulation result of comparison about average length of queue is shown as Figure 8. As our reasoning in the Characteristic of EAI-Based-Grid, the average length of queue in non-grid pattern is longer than that in grid pattern.

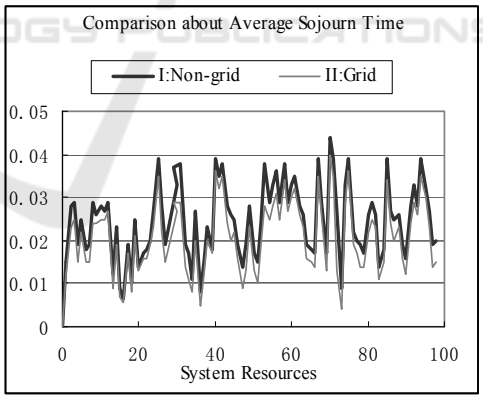

Figure 9: Comparison about Average Sojourn Time.

The Average Sojourn Time W: $W = \lim_{K \to \infty} \sum_{i=1}^{K} \frac{(_{Zi} + _{Si})}{K}$ [Si represents time of receiving service. Zi represents time of waiting].The simulation result of comparison about average sojourn time is shown as Figure 9. As our reasoning in the Characteristic of EAI-Based-Grid, the average sojourn time in non-grid pattern is longer than that in grid pattern.

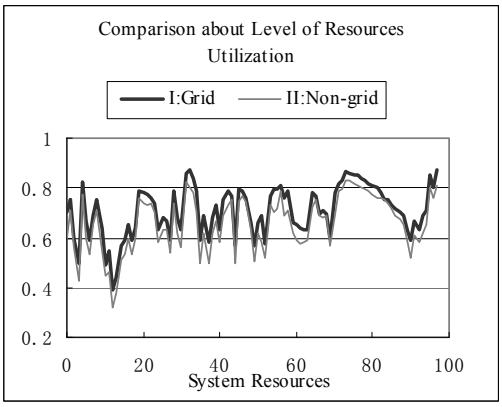

Figure 10: Comparison about Level of Resources Utilization.

The Level of resources utilization u: $\mu = \lim_{T \to \infty} \int_0^T \frac{B(t)dt}{T}$ [B(t) Represents resources running status, busy to take 1,idle to take 0]. The simulation result of comparison about level of resources utilization is shown as Figure 10. As our reasoning in the Characteristic of EAI-Based-Grid, the level of resources utilization in grid pattern is higher than that in non-grid pattern.

From the above results, the pattern based on grid is better than that based on non-grid in the three major indicators .Therefore, the idea of using grid technology in enterprise application integration is feasible in a sense.

6 CONCLUSIONS

Considering the limitations of existing EAI solutions and the advantages of grid technology, we propose enterprise application integration based on grid technology. EAI-Based-Grid solves the problem of existing EAI solutions. First, it provides enterprise with a robust, flexible and sharable IT infrastructure. Second, it achieves cross-platform, cross-boundary and distributed sharing of the enterprise resources. Third, it realizes the loosely coupled integration of enterprise application.

In this paper, we carry out a simulation after we introduce the EAI-Based-Grid solution and its characteristics. The result of simulation proves the solution of enterprise application integration based on grid is superior to non-grid pattern.

However, the grid-related standards are still being continually updated due to its immaturity. Transforming the existing application system into the grid service is still a hard work, which is also the focus of our next step.

REFERENCES

Qureshi K A. 2005. Enterprises Application Integration. *Proceedings of the IEEE Symposium on Emerging Technologies*: 340-345.

Deng Wu, Xinhua Yang. 2008. Study on EAI Based on Web Services and SOA. *2008 IEEE International Symposium on Electronic Commerce and Security*: 95-98.

Ning Zhang, Hong Bao. 2008. Study on Grid Technology and its Application. *4th International IEEE Conference on Wireless Communications, Networking and Mobile Computing*: 1-4.

Xiaoyong Gong, Qingsheng Zhu. 2006. Design and Implementation of Enterprise Application Integration Architecture Based on Grid Technology. *Application Research of Computers, Volume (12)*: 239 -241 (in Chinese).

Yi Gu, Shensheng Zhang, Xiangfei Zhu. 2004. A Study of Enterprise Appication Integration Solution. *Computer Engineering and Applications, Volume (6)*: 209-212 (in Chinese).

Iosup A, Dumitrescu C, Epema D. 2006. How are Real Grids Used? The Analysis of Four Grid Traces and Its Implications. *7th IEEE/ACM International Conference on Digital Object Identifier*: 262-269.

I Foster, C Kesselman, S Tuecke. 2001. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *Journal of Super Computing Applications.*

Benson E, Wasson G., Humphrey M. 2006. Evaluation of UDDI as a provider of resource discovery services for OGSA-based grids. *20th IEEE Interl. Symposium on Parallel and Distributed Processing.*

Bo Wang, Weiqin Tong, Wu Zhang. 2004. Service Organization in OGSI. *The 8th International IEEE Conference on Computer Supported Cooperative Work in Design*: 469-474.