P2P-Based Data System for the EAST Experiment

Yantai Shu, *Member, IEEE*, Liang Zhang, Weifeng Zhao, Haiming Chen, and Jiarong Luo

***Abstract*—A peer-to-peer (P2P)-based EAST Data System is being designed to provide data acquisition and analysis support for the EAST superconducting tokamak. Instead of transferring data to the servers, all collected data are stored in the data acquisition subsystems locally and the PC clients can access the raw data directly using the P2P architecture. Both online and offline systems are based on Napster-like P2P architecture. This allows the peer (PC) to act both as a client and as a server. A simulation-based method and a steady-state operational analysis technique are used for performance evaluation. These analyses show that the P2P technique can significantly reduce the completion time of raw data display and real-time processing on the online system, and raise the workload capacity and reduce the delay on the offline system.**

***Index Terms*—Client-server systems, data acquisition (DAQ), peer-to-peer systems, real-time analysis and display.**

I. Introduction

THE Experimental Advanced Superconducting Tokamak (EAST) [1] is under construction at the Institute of Plasma Physics, Chinese Academy of Science (ASIPP). First plasma is scheduled for late 2005. A computer system - the EAST Data System - is being designed to support the physics research activities by providing diagnostic data acquisition and data analysis. The system is a distributed computer system with a high degree of modularity and capability for autonomous operation.

The previous design for the EAST data system (named HT-7U data system) was based on client/server (C/S) mode [2]. The multiple servers archived acquired data and supported user clients, analyzing data in client/server mode. The system was divided into two basic areas of responsibility: an online system and an offline system. The online system was a data acquisition system responsible for the collection of the data from the EAST experiments. It collected data from diagnostic devices and rapidly transferred large quantities of data to the servers on the offline system. The online system consisted of several independent data-acquisition subsystems, including VXI subsystems, CAMAC subsystems and PCI subsystems. Each subsystem acquired data from several diagnostic devices. The offline system provided data archiving and analysis for the EAST experiment. It consisted of several servers and many PC clients for distributed data processing. The servers had a real-time database and a commercial relational database to store data in different abstraction layers. The real-time database in the servers has a hierarchical structure with several layers. The acquired raw data and processed data are stored in the lowest layer of the real-time database first. The data in each layer of the real-time database are selected and/or concentrated (abstracted) and then put into the layer immediately above. In the highest layer of the real-time database, the data are concentrated (abstracted) and put into the relational database. If the storage is not enough to keep all raw data online, a shift store structure with event trigger will be used for keeping the latest and most meaningful raw data online. Some parts of the previous design for the EAST data system are implemented and used on the existing HT-7 Tokamak for testing.

Peer-to-Peer (P2P) file-sharing applications are a popular way of exchanging files directly between end users across the Internet. Many applications such as Objectivity/DB have been operating in a peer-to-peer environment at Stanford Linear Accelerator (SLAC) and CERN [3]. SLAC is currently using Objectivity/DB to store over 200 Terabytes of online data that is fully distributed over 250 Linux servers. The new design for the EAST data system is instead based on the P2P networking environment, to exploit enhanced capabilities from this configuration. This paper describes the P2P-based EAST data system and evaluates its performance.

This paper is organized as follows: Section II summarizes P2P networking; Section III describes the P2P-based EAST data system; Section IV evaluates the performance of online subsystems; Section V evaluates the performance of offline subsystems; and Section VI includes concluding remarks.

II. P2P Networking

The traditional computing model for many applications is a client/server model. A server computer typically has vast resources and responds to requests for resources and data from client computers. A single server is subject to a single point of failure and can be a bottleneck in times of high network utilization. In the recent years, the evolution of a new wave of innovative network architectures labeled "peer-to-peer (P2P)" has been witnessed. Such architectures and systems are characterized by direct access between peer computers, rather than through a centralized server. P2P networking is the utilization of the relatively powerful computers (PCs) for more than just client-based computing tasks. The modern PC has a very fast processor, vast memory, and a large hard disk, none of which are being fully utilized when performing common computing tasks. The modern PC can easily act as both a client and server (a peer) for many types of applications. P2P networking has more advantages over client/server networking: A network of peers can share its processor, consolidating computing resources for distributed computing tasks, and can allow local resources to be shared directly, without the need for intermediate servers, so

Manuscript received June 9, 2005; revised December 29, 2005. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants 60472078 and 90604013, and the Chinese Academy of Science.

Y. Shu, L. Zhang, W. Zhao, and H. Chen are with the Department of Computer Science, Tianjin University, Tianjin 300072, China (e-mail: ytshu@tju.edu.cn).

J. Luo is with the Institute of Plasma Physics, Academia Sinica, Hefei, Anhui 230031, China (e-mail: jr_luo@ipp.ac.cn).

Digital Object Identifier 10.1109/TNS.2006.874100

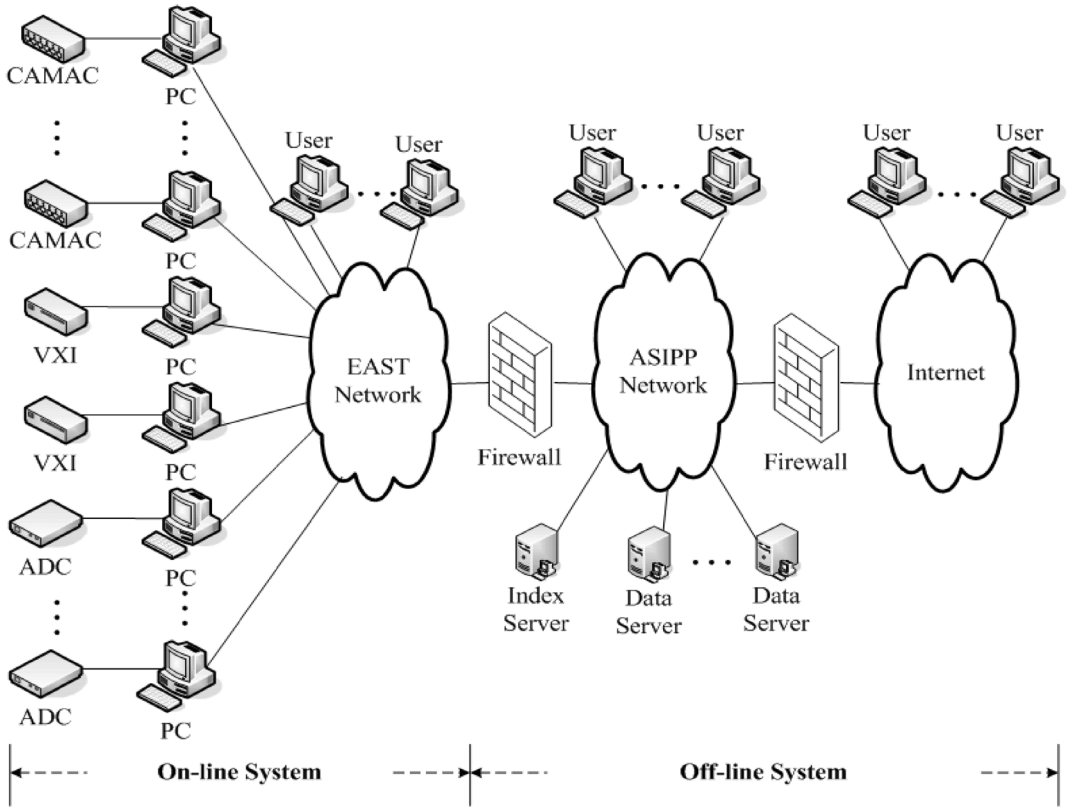

Fig. 1. The EAST data system at the Institute of Plasma Physics.

the whole network system is of low cost. Data/content and resources can be shared from both the center and the edge of the network. A network of peers is easily scaled and more reliable than a single server for avoiding single-point-of failures or performance bottlenecks in the system. As a distributed system, the P2P system needs to deal with nodes joining the system concurrently and with nodes that fail or leave voluntarily and make the system stable. A network of peers allows better traffic-load balancing and allows efficient multipoint communication, so the whole network system has high efficiency.

P2P networking has the following usability features:

- sharing data;
  - — allowing the distribution of multiple types of data e.g., binary, text, audio, and video;
  - — replicated store;
  - — ability to copy data from multiple distributed sources;
  - — ability to search the store using keywords, attributes, and common logical operators;
  - — continuation of partially copied files;
  - — selectable level of compression;
- sharing resources (e.g., CPU, storage);
  - — distributed computing/processing;
- user grouping and naming;
- network load balancing;
- secure communication.

P2P file sharing architectures can be classified by their "degree of centralization", i.e., to what extent they rely to one or more servers to facilitate the interaction between peers. Two categories are identified: purely decentralized and hybrid decentralized [4].

In purely decentralized system, all nodes in the network perform exactly the same tasks, acting both as servers and clients, and there is no central coordination of their activities. Examples of pure decentralized P2P systems are Gnutella [5] and Freenet [6], where every node is a "servent" (SERVers+clieENTS), and can equally communicate with any other connected node. Each user has to distribute queries from node to node. It is hard to find the desired files without distributing queries widely. Therefore, the searching time is rather long and fluctuates sharply. It also introduce large mount of query packets.

However, the most widely used file-sharing systems, such as Napster[7], do not fit this definition because some nodes have special functionality. For example, in Napster, a server node indexes files held by a set of users. Users search for files at a server, and when they locate a file of interest, they download it directly from the peer computer that holds the file. We call these types of systems hybrid because elements of both pure P2P and C/S systems coexist. Currently, hybrid file-sharing systems have better performance than pure systems because some tasks (like searching) can be done more efficiently in a centralized manner, e.g., locate files quickly and efficiently.

III. EAST Data System

A simplified block diagram of the EAST data system is shown in Fig. 1. The EAST data system is based on a mesh network and is connected to the ASIPP's laboratory network and the Internet

through a firewall. The EAST data system includes an online system and an offline system.

The online system consists of several independent data acquisition subsystems, including four CAMAC subsystems, two VXI subsystems and five PCI subsystems. They are responsible for the collection of the data from diagnostic devices. References [2] and [8] already describe these data acquisition subsystems in detail. To provide more rapid access to acquired raw data after a shot, all collected data are stored in the data-acquisition subsystems locally as primary raw data archiving (online distributed real-time database), and the users on the online system can access the raw data directly using P2P mode; this configuration provides more rapid access to large quantities of data than by having users wait for data to be transferred to the servers on the offline system. When the EAST operates in pulse mode, the raw data for one experiment shot consist of thousands of meta-data; for example, data acquired from one channel during one experiment shot could be a meta-data which consists of a part of data and a header which includes data calibration parameters and data acquisition configuration information and so on.

The P2P architecture of the EAST data system is a Napster-like architecture, EastNap, which extends the Napster protocol to allow sharing of data in a real-time database and a relational database. The first, and perhaps best known, P2P, Napster, is a protocol for sharing MP3 (music) files between users. With Napster, users who wish to download the file first query the index server for the location of the file, then download directly from the file owner's computer. The Napster architecture was integrated with searching performed centrally at the Napster server and downloading decentralized directly between two users. The EastNap uses structural filename which makes the query easier. To improve the searching performance on the EastNap index server, the index content resides in memory as the index server handles only a very small amount of index data. Therefore the EastNap has simplicity and fast query response advantages, and it is easy to implement sophisticated search engines on top of the index system. The EastNap can store over tens of Terabytes of data that is fully distributed over hundreds of PCs. To overcome single points of failure and improve reliability, the EastNap has the ability to link several index servers together.

The offline system provides secondary data archiving and data analysis for the EAST experiment. It consists of a few data archiving servers and many PC clients for distributed data archiving and processing. All collected data are transferred to the archiving servers on the offline system after the busy period of the data-acquisition subsystems. Then the PC clients can get the raw data from the archiving servers. The PC clients have an offline real-time database and a relational database to store data in different abstraction layers. The P2P architecture of the offline system is a Napster-like architecture, EastNap, too. With P2P computing, each participating computer (PC client), referred to as a peer, functions as a client with a layer of server functionality. This allows the peer to act both as a client and as a server within the context of a given application. P2P applications build on such functions as storage, computations, messaging, security, and file distribution, when handled through direct exchanges between peers. A peer can initiate requests, and it can respond to requests from other peers in the network (inside and outside ASIPP). The ability to make direct exchanges with other users liberates P2P users from the traditional dependence on central servers. Users have a higher degree of autonomy and control over the services they utilize.

The offline system uses a distributed database with a hierarchical structure with several layers. First after the experiment shot, the raw data are transferred into the lowest layer of the real-time database in the servers. Then each participating computer (PC client) on the offline system gets a part of the raw data from the real-time database in the servers and maintains a replicated store (the secondary real-time database); as it becomes available, the processed data are put into the real-time database in PC clients as well. The data in each layer of the real-time database are selected and/or concentrated (abstracted) and then put into the layer immediately above. Finally, the concentrated (abstracted) data are put into the relational database in servers and PC clients at a later time.

All users on the offline system are divided into groups. Group members maintain a replicated store containing all the shared data of the group and can search the store using keywords, attributes, and common logical operators.

IV. Performance Evaluation of the Online System

There are three kinds of users in the EAST data system: operation engineers, operation physicists and research physicists. The operation engineers and operation physicists work on the online system, and research physicists work on the offline system. There are three workloads on the online system after each shot:

- — real-time display of 200 channels of raw data interesting to the operation engineers on 10 PC clients,
- — real-time processing of 400 channels of data interesting to the operation physicists on 40 PC clients and
- — transferring of 2000 channels of data (3.2 GB) from the data acquisition subsystems to the servers where the research physicists perform offline analysis.

Here real-time implies more of a pseudo-real-time response and representative of between-shot time-scales as needed for analysis and display of data.

Performance evaluation is very important throughout the development of a real-time system. Performance analysis, simulation and measurements should be used together to calibrate and validate each other because premature actions may be taken based on invalid assumptions, incorrect data, or hidden modeling errors. We have used analysis and simulation models similar to those in [9] and [10] to evaluate the performance of the EAST data system. An approximation analysis of the online system using steady-state operational analysis method has been used to model the data flows and predict the completion time because the workloads on the online system are deterministic. In addition to the analysis, we did many RTSS simulations [9] of the online system. The RTSS is a flexible event-driven simulation software. Since 1990, the RTSS simulator has been developed and used for modeling the distributed data acquisition and processing systems at JET (JET Joint Undertaking, UK) and

TABLE I
ANALYSIS RESULTS OF THE C/S BASED ONLINE SYSTEM

System	Data collection time (s)	Data available time on servers (s)	Completion time of raw data display and real-time processing (s)
CAMAC DAQ	7.9		
VXI DAQ	35.2		
PCI DAQ	0		
C/S with 1 server		157.3	285.1
C/S with 2 server		87.3	151.2
C/S with 4 server		58.2	91.7

TABLE II
ANALYSIS RESULTS OF THE P2P BASED ONLINE SYSTEM

System	Data collection time (s)	Data available time on DAQ subsystems (s)	Completion time of raw data display and real-time processing (s)
CAMAC DAQ	7.9		
VXI DAQ	35.2		
PCI DAQ	0		
P2P with 11 DAQ		35.2	68.7

ASIPP. The simulation results are approximately the same as those provided by a steady-state analysis.

The results of analysis modeling for the online subsystem in the C/S environment are shown in Table I. From the end of the EAST pulse, the raw data will be available on all data-acquisition subsystems at 35.2 second. The raw data on the servers will be available at between 157.3 and 58.2 seconds for 1 or 3 servers. The completion time of raw data display and real-time processing will similarly range from 285.1 to 91.7 seconds. The results of analysis modeling for the online subsystem in the P2P environment are shown in Table II. In the P2P environment, from the end of the EAST pulse, the raw data will be available on the data-acquisition subsystems at 35.2 second. With the PC clients accessing the raw data on the data-acquisition subsystems directly using P2P mode, the completion time of raw data display and real-time processing will be at 68.7 seconds if the online system has 11 data-acquisition subsystems.

Comparing the P2P online system with the C/S online system, the analyses show that a P2P online system can significantly reduce the completion time of raw data display and real-time processing. The main reason is that the users (operation engineers and physicists) can access the raw data on the data-acquisition subsystems directly and more quickly using P2P mode instead of accessing data on the servers on the offline system. The P2P system distributes the reading-data job to more DAQ computers (PCs). Analysis of the EastNap index server shows that it will not affect the above performance evaluation result of the online system.

V. PERFORMANCE EVALUATION OF THE OFFLINE SYSTEM

There are two workloads on the offline system: 1) the data-processing workload from the research physicists working on the offline system, and 2) the data transfer from the online system to the offline system. The requirements of the research physicists make up the main workload of the offline system. In contrast with the workloads of the online system, these are probabilistic. That is, the majority of the complex operations that take place on the offline system can be described by means of macroscopic probabilistic assumptions, neglecting the actual behavioral details. Several RTSS models of the offline system have been used to simulate the data flows and predict the delay and utilization factor on the data servers, the index server and user clients. Some models of the offline system are based on the C/S environment while other models of the offline system are based on the P2P environment.

When a research physicist is working on the offline system, the workload includes three parts: 1) getting one or more channel's data from the database, 2) processing or computing on the data, and 3) plotting the results of the processing. In the C/S environment, only the first part (fetching the data) requires the server capability, while part 2 (processing the data) and part 3 (plotting the results) in the above workload use the client capability. The queuing theory [11] involves a three-component description, A/B/m, which denotes an m-server queuing system where A and B "describe" the interarrival time distribution and service time distribution, respectively. A and B take on values from the following set of symbols, which are meant to remind the reader which distributions they refer to: M = exponential (i.e., Markovian), D = Deterministic, and G = General. For approximation analysis, omitting the effect of data transfer from the online system we build an M/M/1 queue model for the data server on the offline system under the C/S environment. In the model, a queuing facility represents the server capability and customers represent the user's workloads. To estimate the model parameters, some measurement experiments are done on the C/S environment testbed (with 2 GHz CPU, 30 MB/s disk reading speed, 1 Gbps Ethernet, and 1.5 MB/channel average data). The disk-driver-call overhead, network-driver-call overhead, disk-read I/O and other parameters are measured. According to the analysis results of measurement experiments on the offline system, we assume that the arrival stream is a Poisson process and the service-time distribution is of exponential form. The average service rate is 4.76 channel/s. Solving the M/M/1 queue model, we can get the average delay versus average workload (by the number of channels/second) on the system and so on. The results of analysis modeling for the servers on the offline system under the C/S environments are shown in Fig. 2. If the system has one, two or four servers, the server's performance is shown by three curves of the average delay versus workload capacity. We suppose that research physicists will not like the delay larger than one second. Under the C/S environment, the workload capacity can be 3.5, 8.5 and 18 channels/sec respectively.

In our P2P environment, on startup, the user contacts the index server and sends a list with the data it maintains. When the server receives a query from a user, it searches for matches

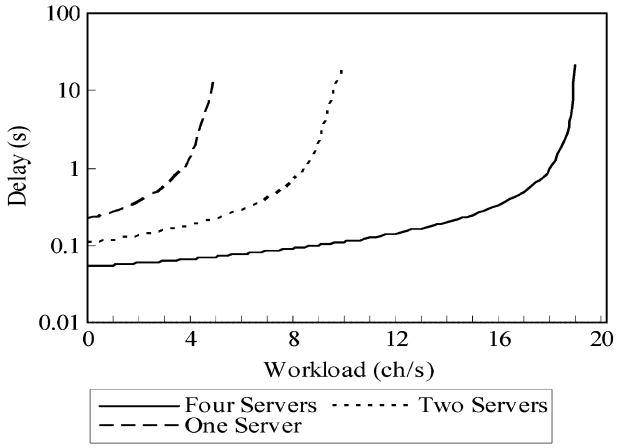

Fig. 2. Performance of the servers on the offline system under the C/S environment.

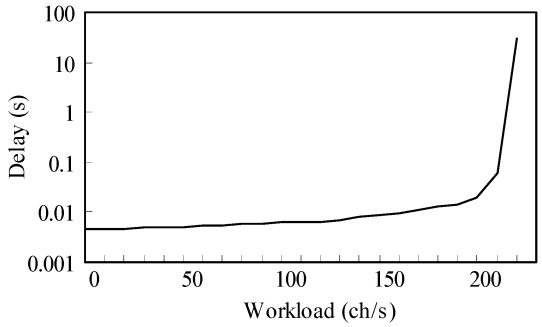

Fig. 3. Performance of the EastNap index server under the P2P environment.

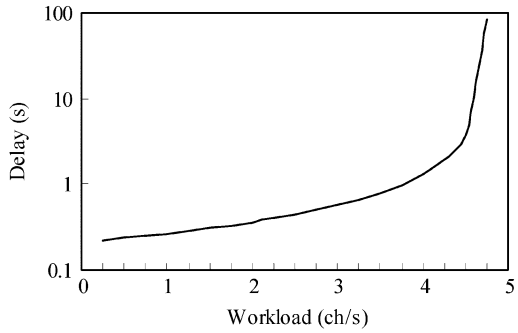

Fig. 4. Performance of the PC peer (as data source) under the P2P environment.

in its index, returning a list of peers/users that hold the matching data. The user then opens a direct connection with the peer that holds the requested file, and downloads it. In comparison with the C/S computing environment, the P2P computing paradigm needs only do a searching on the index server. In the P2P environment, for the time being, we do not take I/O costs into consideration, but assume that memory is cheap and all indexes may be kept in memory. The index server transmits only the searching result, not the data itself, to user clients. In the C/S environment, however, the server has to transmit the data whose size is far more than that of the searching result. Therefore the P2P computing paradigm needs much less index server time for searching only one channel.

As discussed above, the workload of an index server should contain three scopes of work: 1) receiving the query from the users, 2) searching for the query, and 3) sending the results to the users. In the P2P environment, as the searching time and the time for transmitting the searching result is near constant, we build an M/D/1 queue model for the index server on the offline system. To estimate the model parameters, some measurement experiments are done on the P2P environment testbed (with 2 GHz CPU, 1 Gbps Ethernet). The CPU-searching-time, network-driver-call overhead and other parameters are measured. According to the analysis results of measurement experiments on the offline system, we assume that the arrival stream is a Poisson process. The constant service rate is 217.4 channels/sec. Solving the M/D/1 queue model, we can get the average delay, utilization factor on the system and so on. The results of analysis modeling for the index server on the offline system under the P2P environment are shown in Fig. 3 by the curves of the average delay versus workload capacity (by the number of channels/second). Under the P2P environment, the workload capacity can be raised to 215 channels/sec.

In addition, we build an M/M/1 queue model to evaluate the performance of the PC peers which act as data servers on the offline system under the P2P environment. The results are shown in Fig. 4.

Both simulation and analysis studies indicate that in comparison with C/S systems, a P2P system can significantly raise the workload capacity and reduce the average delay of the offline system. There are two main reasons. One is that the index server transmits only the searching result and not the data itself in the P2P environment. In the C/S environment, however, the server has to transmit the data whose size is far larger than that of the searching result. The P2P system distributes the reading and transferring data job to client computers (peers). The number of the clients is much more than the number of the servers on the offline system. Thus the heavy load on the servers is distributed to many peers, and load balance is implemented on the P2P based offline system. The other reason is that as the index server maintains only the index which is a very small mount of data; all the index can be kept in memory, thereby keeping the searching time needed for a query on the P2P system quite small.

VI. Conclusion

The new design for the EAST data system is based on Napster-like P2P architecture. Instead of transferring data to the servers, all collected data are stored in the data acquisition subsystems locally and the PC clients can access the raw data directly using the P2P architecture. This allows the peer (PC) to act both as a client and as a server. Its main features can be summarized as follows: sharing data; sharing resources (e.g., CPU, storage); low cost; and network load balancing.

A simulation-based method and a steady-state operational analysis technique are used for performance evaluation. The simulation results are approximately the same as those provided by a steady-state analysis. Comparing the P2P-based online system with the C/S based the online system, the analysis researches show that a P2P-based online system can significantly reduce the completion time of raw-data display and real-time processing. Both simulation and analysis studies indicate that,

in comparison with C/S based offline systems, a P2P-based offline system can significantly raise the workload capacity and reduce the delay.

To try to get a better idea of the impact of P2P on performance we are working to modify and install a P2P testbed on the existing HT-7 tokamak data system [12]. Future work includes doing more detailed and accurate measurements on various subsystems and networks, building more P2P testbeds, and applying simulation and analysis techniques to optimize the P2P-based EAST Data System.

ACKNOWLEDGMENT

The authors would like to thank their colleagues in the Computer Science Department, Tianjin University, and the Computer Division, Institute of Plasma Physics Academia Sinica, for their contributions. They are also grateful to their friend Dr. N. R. Sauthoff for his useful corrections and comments.

REFERENCES

[1] P. Heitzenroeder, Fusion Technology Committee Rep., San Diego, CA, Oct. 2003 [Online]. Available: http://www.ieee.org/organizations/pubs/newsletters/npss/0304/fusion.html

[2] Y. Shu, J. Luo, F. Zhao, H. Wang, and D. Wang, "Performance evaluation of the HT-7U data system," *IEEE Trans. Nucl. Sci*, vol. 49, no. 2, pp. 428–431, Apr. 2002.

[3] Objectivity joins peer-to-peer (P2P) standards body, Objectivity, Inc., Sunnyvale, CA [Online]. Available: http://www.objectivity.com/News/PressReleases/2001/News_PR_013101_GGF.shtml

[4] B. Yang and H. G. Molina, "Comparing hybrid peer-to-peer systems," in *Proc. 27th Int. Conf. Very Large Data Bases*, Rome, Italy, Sep. 11-14, 2001, pp. 561–570.

[5] Gnutella Development Home Page [Online]. Available: http://gnutella.wego.com/

[6] Freenet Home Page [Online]. Available: http://freenet.sourceforge.com/

[7] Napster Home Page [Online]. Available: http://www.napster.com/

[8] Y. Shu, J. Luo, J. Yan, F. Zhao, and L. Zhang, "PCI/IRMX-based front-end data acquisition for the ht-7u experiment," *IEEE Trans. Nucl. Sci.*, vol. 51, no. 3, pp. 420–424, Apr. 2004.

[9] Y. Shu, M. Jia, Y. Fei, Y. Zhang, G. Liu, S. Yang, and Y. Chen, "Progress on RTSS simulation-based analysis for real-time systems development at two laboratories," *IEEE Trans. Nucl. Sci.*, vol. 43, no. 1, pp. 74–78, Feb. 1996.

[10] Y. Shu, Z. Jin, and G. Liu, "Modeling of HT-7 data processing system," in *Proc. 10th IEEE Real-Time Conf.*, Beaune, France, Sep. 22-26, 1997, pp. 527–530.

[11] Leonard Kleinrock, *Queueing Systems Volume 2, Computer Applications*. New York: Wiley, 1976.

[12] J. Luo, H. Wang, Z. Ji, L. Zhu, F. Wang, and Y. Shu, "The distributed control and data system in HT-7 Tokamak," *IEEE Trans. Nucl. Sci*, vol. 49, no. 2, pp. 496–500, Apr. 2002.