"Grid Computing as Applied Distributed Computation: A Graduate Seminar on Internet and Grid Computing"

James C. Browne
Department of Computer Sciences
University of Texas
browne@cs.utexas.edu

Abstract

Internet and Grid based systems, whether their purpose is computation, collaboration and information sharing, are all instances of systems based on application of fundamental principles of distributed computing. This paper reports on a graduate seminar on Internet and Grid computing which focuses on the fundamental principles and concepts of distributed computation which underlie Internet and Grid systems, both software and applications. The course takes a broad definition of Internet and Grid computing, ranging across high performance computation to peer-to-peer file sharing systems. The focus is on identification and analysis of the concepts and principles underlying grid computing and how they are realized and applied in systems ranging from Globus to Freenet with web services in between. In summary, the course is a "what is it" rather than a "how to do it." Each participant does a project concerning an Internet or Grid computing subject and presents it to the class.

1. Motivation and Overview

Humans, computers and communication systems are gradually being synergistically coupled to create systems for sharing resources of all types with goals including analyzing and solving problems, implementing collaborations and sharing information. Internet and Grid based computation and collaboration and sharing systems are instances of such systems.

The seminar takes a very broad definition of what constitutes an "Internet or Grid" system. Any collection of resources which is organized together through a communication network to pursue some goals is taken to constitute an "Internet or Grid" system. The systems analyzed and studied in this seminar range from full-fledged Grid infrastructures such as Globus toolkits [1] and Legion [2], to programming systems for Grid applications such as Triana [3] or ICENI [4], to composition systems for web services, to distributed file systems and peer to peer network file sharing systems.

All of these systems are based on upon fundamental principles and algorithms of distributed computing. These fundamental principles constrain the design and development of these systems but their influence is not always recognized and understood in the Grid systems research community. One goal of this class is to understand how the fundamental principles and concepts of applied distributed computing are applied and realized in Internet and Grid computing systems. It is also based on the belief that the best and most effective way to understand these principles and their applications is to analyze, evaluate and use the current emerging systems for Internet and Grid computing.

There are also several research communities pursuing topics and developing systems related to and grid computing. Another goal of this course is to establish the relationship among all of these research communities and to look for synergisms and possible mutual benefit.

The prerequisites for this seminar are an understanding of the fundamental principles of distributed systems such as would be obtained from a senior undergraduate or a first year graduate course on distributed systems.

The target audience for the seminar is students in research in an area related to grid computing and students considering research in grid computing or a related topic. The majority of students in the first two offerings of the seminar have been students already in research in a broadly related area. There have also been a substantial number of students who are employees of high-tech companies with facilities in Austin.

The course begins with a sequence of lectures which present fundamental principles and algorithms and example systems. Each student executes a project which covers some topic in internet and grid systems and gives two presentations to the class. This course has been offered twice, in the Fall of 2002 and in the Fall of 2003. There were 20 students in the Fall of 2002 and 14 students in the Fall of 2003. The Fall 2002 offering produced one published paper [5] and a second [6] will follow. A similar result is expected from the Fall 2003 semester offering.

Section 2 sketches some of the concepts upon which the analyses of Grid systems are based. Section 3 sketches the lecture topics and Section 4 sketches the

0-7803-8430-X/04/$20.00 ©2004 IEEE

projects done by seminar participants. Sections 5, 6 and 7 cover course materials, course evaluation and comparison to other courses with conclusions.

2. Applied Distributed Systems

This section summarizes the perspective used in the lectures for analysis and classification of internet and grid systems as distributed systems. The focus issues are system definition and system control including fault-tolerance and the programming model for creating instances (applications) of the internet or grid distributed system. The lectures and the projects focused on understanding how these issues are instantiated in Internet and Grid systems.

2.1 System Definition and Control

Definition: Distributed System

A distributed system is a collection of logically or physically disjoint entities which have established a process for making collective decisions.

Collective decisions depend on common knowledge held by the participants in the decision process. Distribution intrinsically leads to uncertainty in system state. Uncertainty can be bounded through use of interaction protocols which narrow the time span of uncertainty and/or the elements of system state which are uncertain.

The fundamental requirements for implementing a distributed system are a capability for creation of a system and a capability for control of the system through decision processes. The algorithms which can be used in the decision process depend upon the properties of the computer, communication and data resources which comprise the distributed system and in particularly upon the assumed reliability of the computer and communication resources.

Grid systems for instantiating distributed systems (called virtual organizations) implicitly assume some reliability properties for the resources they assemble into a virtual organization but often these assumptions are either those of complete reliability or are not explicitly stated.

Definition: Decision Algorithms

Decisions can be abstractly formulated as

If (Decision(Current State, Request))
then

State =Transition(Current State, Request);

A decision is a specification for execution of a change of state. *Decision* is a function which evaluates to true if a Request for a state change is to be accepted. *Transition* is a function which transforms the current state to a new state.

The analyses of grid systems used in the lectures targets identification of the assumptions made with respect to reliability and characterization of the decision algorithms.

2.2 Central versus Distributed Control

Ideally the state of the system to which *Decision* is applied is complete and accurate and *Decision* is also a complete function. This is straightforward in a single site system where the system state can be maintained in a consistent state in a local data structure and *Decision* applied to this data structure. Distributed control implies that *Decision* is partitioned among the entities composing the system and coupled by communication protocols. Central control of a distributed system can, however, be implemented by gathering the system state at a single site, executing *Decision* at this site and propagating the decision to the other sites. In fact, this is what is normally done in most Grid systems. Execution of the *Transition* function is, however, intrinsically distributed. This leads to intrinsic uncertainty in the state since a resource can fail during the time span of making the decision and executing the state transition. Internet and grid systems should make provision for dealing with this uncertainty.

Distributed control implies that each entity makes decisions following a commonly agreed upon process and based upon agreed upon common knowledge. An important aspect of a distributed system which utilizes distributed control is specification of the commonly agreed upon processes and the common knowledge upon which distributed control is based. These aspects should be precisely specified for Grid systems which utilize distributed control.

Maintenance of an instantaneously updated and consistent state for a distributed system requires attainment of continuous consensus. Attainment of consensus is a classical problem of distributed systems [7,8]. Consensus can only be obtained for a specific set of execution environments.

The analyses of grid systems done for the lectures include characterization of the model for control, specifications for the commonly agreed processes, specifications for and mechanisms for maintaining common knowledge and the conditions under which consensus can be attained under the assumptions made for reliability and for the decision algorithms which are specified.

2.3 Programming Model

Grid software systems are commonly presented as first creating a transient "virtual organization" or virtual resource configuration, creating an application to execute on the virtual resource configuration and then executing the application. Therefore most Grid software systems implicitly or explicitly incorporate a programming model. The programming model determines the properties of the application. It is, however, not always the case that the papers describing the systems explicitly defined or specified a programming model. Additionally programming models incorporate assumptions concerning the environment in which the programs will execute. These assumptions are, as previously noted, not always explicitly specified for Grid systems.

A key goal for the lectures and class discussions was to analyze the systems to extract and make explicit the underlying programming model. The programming model is usually based on extending some existing parallel programming model such as a control flow model or a data flow model, etc. Important issues to be considered in extending conventional parallel programming models for grid environments include whether uncertainty of system state is visible in the programming model and whether dynamic program structures are supported.

3. Lecture Topics and Coverage

The instructor gives 15-20 lectures on basic topics concerning characterization and categorization of Internet and Grid computing systems and implementation of internet and grid systems. The lectures use a broad definition of internet and grid computing. The lectures cover systems which are labeled grid systems such as Globus [1], Legion [2], Tiana [3] , ICENI [4], G2 [9], Harness [10] and X-Cat [11] but also cover web services and multiple implementations of peer to peer systems. The first two lectures are on basics of distributed systems technology and the remaining lectures are focused on analysis of one or more example systems. Each example system was analyzed for its properties as a distributed system including naming models, mechanisms for attaining consensus, mechanisms for system control, programming model, provision for fault-tolerance, security, etc. The lectures on example systems are based on papers which are circulated in advance. The lectures on example systems are given partly by the instructor, partly by the students and by guest lecturers from companies with Grid-oriented products or services. The instructor gave the first six-eight lectures on example systems and the students gave the last three or four lectures. Each student is asked to prepare for each system an analysis which covering the issues listed above and a summary set of slides on the paper or papers. Students were chosen at random to offer their presentations and analyses of these papers. The third set of lectures which covered approximately 12 lectures were presentations by students on their projects. The set of example systems chosen for class coverage is adjusted for the coverage which will occur as parts of the project presentations. There are two presentations on each project, a first presentation which posed the problem being pursued under the project and a second which gave the results of the project.

Table 1, which follows, gives a list of the lectures given by the instructor and the lectures on some of the example systems.

Table 1 – Lecture Topics

Title	Content Summary
1. Fundamentals of Distributed Systems	Basics of distributed systems, consensus, common knowledge, fault tolerance and programming models.
2. Introduction to Grids	Motivation for grid systems, resource sharing, virtual systems and virtual organizations.
3. Characterization of Grids	Properties, characterization and classification of grid systems. This lecture is partly based upon the papers by Nemeth and Sunderam [12], presents the taxonomy provided by these authors supplemented with additional material from [13].
4. Introduction to Peer to Peer Systems	Mechanisms for distributed control, use of distributed hash tables [14] as a routing mechanism and similarity-based routing. Systems discussed include: CAN [15], Chord [16], Pastry [17] and Freenet [18].
5. Web Services	Overview of Web services - XML, Soap, WSDL and UDDI as the basis of Web services.
6. Web Services	Composition of web services into systems – data flow and control flow concepts, WSFL and BPE4Laws.
7. Globus	Structure of and mechanisms provided in Globus 2 with simple example programs.
8. Legion	Basics of Legion [2] including naming model and programming

	model.
9. Physiology and Anatomy of the Grid – OGSI/ODSA	This lecture sketches the content of the two papers [19,20] defining the emerging grid standards
10. Desktop Grid computing	This lecture was given by a representative of United Devices on its commercial implementation of the seti@home desktop grid model
11. Key word search in peer to peer systems	Survey of algorithms and mechanisms for keyword search in peer to peer networks including distributed hash table based systems [21], [22], vector space search [23]] and combining distributed hash table and vector space searches [24]
12, 13 and 14. Grid Programming Systems	Survey of Grid programming environments including Triana [3], G2 [9], X-CAT [11], ICENI [4], Armada [25] and Harness [10].
15. Performance Issues	Performance issues for parallel programs and parallel computations implemented on grid infrastructures.
16. Coordination Models	Survey of coordination models and languages. (Coordination models and languages focus on composition of distributed systems by coordination among systems of distributed processes.)
17. Microsoft .Net	Microsoft's .Net Web services development environment
18. Commercial Issues for Grid Infrastructures	A guest lecturer from IBM-Austin's local laboratory of commercialization of Grid software. It covered primarily reliability and security.
19. Peer to Peer Database Systems	Qualification, retrieval and update with distributed control added to distribution of data. Systems used as illustrations include [26] and [27].

4. Student Projects

Projects could be undertaken by an individual or a team of two persons. The instructor provides a list of possible projects but encourages each student, particularly those who are engaged in research, to formulate a project which connects his or her research to Internet and Grid computing. Several of the students in the class were working in the area of distributed systems or Internet and Grid computing and did propose topics for their projects. Projects play an important role. The instructor interacts with the students to plan the projects. Project presentations are structured to incorporate coverage of topics of general interest. For example, the lectures on JXTA and Grid resource management were done as parts of project presentations. The titles of the projects, which are fairly descriptive, are given in Table 2 along with some comments on what was covered in the reports on the projects. The projects which involved actual grid computations were able to use the computational and storage resources of the Texas Advanced Computation Center (TACC, http://www.tacc.utexas.edu) and the resources of the Computer Sciences Department (CSD) and an internal grid incorporating resources in TACC and CSD. Project reports are available from the class web page.
http://www.cs.utexas.edu/users/browne/cs395f2003/

Table 2 – Student Project Titles and Summaries

Project Title	Abstract
A Taxonomy For Fault-tolerant, Reliable Distributed File Systems	This project is a survey and analysis of four secure fault-tolerance distributed file systems: Farsite, OceanStore, Ivy, and Frangipani. Each file system is analyzed with respect to fault-tolerance, scalability, usability, maintenance overhead, and consistency. A taxonomy for distributed file systems based upon their failure models, update mechanisms, and data location schemes is developed and applied to the four systems studied.
A Formal Model for Incorporating Performance Guarantees in Grid-based Systems	This project designed and evaluated an approach to introduction of Quality of Service (QoS) guarantees within OGSA architected Grid systems. The project report defines and describes the components for building a QoS capability for Grid systems and discusses the concepts of reliability, pricing models and techniques to guarantee delivery. The system design was evaluated by simulation.
Locality-based Multi-writer Replication of Protocols for Internet-scale Systems	This project designed and evaluated by simulation a locality based replication protocol that combines the benefits of existing protocols. This protocol, which takes advantage of read/write locality, is shown to scale well under workloads with all read-to-write ratios.

A Taxonomy of Grid Resource Management Systems	The resource management system is the central component of a Grid system. This project surveys the resource management architectures of popular contemporary Grid systems, defines and describes a taxonomy for classifying resource management architectures and applies the taxonomy to the systems studied
Peer to peer Data Sharing on a Network of Labview Nodes	The goal of this project was to overlay a peer-to-peer (P2P) data grid on a set of LabVIEW (LV) nodes. LV nodes, in this context, are data acquisition sensors that access live data. These nodes are also capable of running embedded LabVIEW Real-Time (LVRT) applications, which allows creation of a P2P network with these nodes by creating a global namespace for these sensors such that a client can use a logical name of the sensor and query the P2P grid to locate the sensor.
Solving Quarto with JXTA: Peer to Peer Game Playing	This project designed and implemented a peer to peer grid-based autonomous quarto player. The implementation uses the JNGI framework which itself is written on top of JXTA.
XML Matching in Peer to Peer Systems	This project formulated an automata based approach for matching XML documents with user profiles. This system is based upon two components, XFilter which can extend information filter functionalities in terms of expressiveness of user profiles and matching effectiveness and a p2p networking environment where peers publish user profiles to collect and filter XML documents. Combining these two technologies yields an XML filtering system for p2p networks in which peers in the networks are allowed to share XML documents and each peer can express its interest to select XML documents.
Writing Grid Services Using the GT3 Core	This project was a demonstration and evaluation of the GT3 core for building Grid Service. The report presents a step-by-step procedure for building Grid Service based applications
Implementation of the GAML Algorithm on a Desktop Grid	This project attempted a SETI@home implementation of a "tree of life" application. The goal was to evaluate the effectiveness of this model of computation for this family of algorithms.
Supporting Range Queries in a Distributed System	The goal of this project was to design and build a scalable distributed discovery system for documents that (i) supports both simple queries and range queries on document names, (ii) supports efficient insertion and deletion of documents, (iii) distributes both storage and access loads uniformly among all the participants, and (iv) is efficient in terms of the communication cost incurred for responding to queries. The project was carried through design and design evaluation with demonstration of effective load balancing.

5. Course Materials

The course materials are papers, web pages, and specification documents. There are far too many to include citations in this paper but the list of papers, document and urls can be found on the class web page. Some of the lectures are available on the web page as pdf files but many of the lectures were informal and/or taken directly from the papers. Citations to references for individual systems have not been included in the references for this paper because space does not permit. But the full set of references are included on the class web page.

6. Related Research Areas

In addition to the broad area of distributed systems there are other research communities approaching the problems of implementing transient virtual systems from distributed resources. The coordination models and languages research community focuses on models and languages for coordinating systems of distributed processes [28]. Since resources are ultimately represented by processes in implementation there is a direct mapping between the problems and solutions originating in the two research communities.

Distributed databases is another related research area. Gribble, et.al. [29] is a representative paper exploring the intersection between distributed databases and peer to peer systems.

7. Course Evaluation

There are several metrics for evaluation of courses. The student evaluations for this seminar were among the highest for any class ever taught by the instructor. Another metric for a research seminar class is publication of research initiated in the seminar. At least two papers will result from the Fall 2002 offering and others may still result. If there is sufficient follow through then there should be two more papers published from research motivated through the Fall 2003 offering. The results from the two projects "Supporting Range Queries in a Distributed System" and "Locality-based Multi-writer Replication of Protocols for Internet-scale Systems" will become segments of the Ph.D. dissertations of the students who executed these projects and both have the content and substance to justify submission for publication.

Additionally positioning internet and grid computing research in the context of other related research topics has generated insights into to how to apply and use results from these research areas in internet and grid computing research.

8. Comparison to other Courses and Conclusions

The instructor has benefited greatly from the instruction materials available on the web pages for other classes including classes at ISI, USCD, University of Houston, University of Alabama at Birmingham, Rutgers, University of Chicago, SUNY-Binghamton, University of Surrey in the UK, University of Guelph in Canada and KTH in Sweden. A great deal of material has been taken from these sources. But none of these courses takes a perspective as analytical as this course. The "what is it" approach characterizing internet and grid systems not as a new topic but rather as an application of principles and concepts of distributed systems is different from most of the other courses and seems a viable approach. This approach seems appropriate for a course offering in a computer science program where design, analysis and understanding are equally as important as operational skills in implementation. Those students interested in "how to" mostly chose projects involving implementations and system evaluations and had in-depth experience with use of internet/grid infrastructures and development environments.

9. References

[1] I. Foster and C. Kesselman, " Globus: A Metacomputing Infrastructure Toolkit" ***Intl J. Supercomputer Applications***, 11(2):115-128, 1997.

[2] M. Lewis, et. al."Support for extensibility and site autonomy in the Legion grid system object model". Journal of Parallel and Distributed Computing 63(5): 525-538 (2003).

[3] I. Taylor, M. Shields, I. Wang and R. Philp.Distributed P2P Computing within Triana: A Galaxy Visualization Test Case, *Proceedings of IPDPS 2003* (April 2003. Nice, France).

[4] J. Hau, W. Lee and S. Newhouse, "Autonomic Service Adaptation in ICENI using Ontological Annotation" *Proceedings of 4th International Workshop on Grid Computing, Grid 2003* (Phoenix, AZ, Nov. 2003).

[5] K. Sankaralingam, S. Sethumadhavan, and J.C. Browne, J.C., "Distributed Pageranks for P2P Systems" *Proceedings of the Twelfth IEEE International Symposium on High Performance Parallel and Distributed Systems* (2003) 58-69.

[6] K. Sankaralingam., M. Yalamanchi, and Browne, J.C., "Distributed Computation of Page Ranks: Numerical Properties and Embedded in Web Servers" (submitted to a special issue of the *Journal of Grid Computing* devoted to papers from HPDC12).

[7] J. Turek, and D. Shasha, "The Many Faces of Consensus in Distributed Systems," *IEEE Computer*, Vol. 25, Issue 6, pp. 8-17, 1992.

[8] D. Dolev, C. Dwork and L. Stockmeyer, "On the Minimal Synchronization Needed for Distributed Consensus," *Journal of the ACM*, Vol. 34, No. 1, pp. 77-97, 1987.

[9] W. Kelly and L. Frische, "G2 Remoting: A Cycle Stealing Framework based on .NET Remoting", *Proceedings of the 2003 APAC Conference on Advanced Computing, Grid Applications and eResearch*, (Gold Coast, September 2003).

[10] M. Migliardi, D. Kurzyniec and V. Sunderam "Standards Based Heterogeneous Metacomputing" The Design of Harness II" *Proceedings of IPDPS'02* (Ft. Lauderdale, FL., April 2002).

[11] M. Govindaraju, S. Krishnan, K. Chiu, A. Slominski, D. Gannon, and R. Bramley. "XCAT 2.0: A Component-Based Programming Model for Grid Web Services" (Technical Report-TR562, Department of Computer Science, Indiana University. Jun 2002).

[12] Z. Nemeth, and V. Sunderam, "Characterizing Grids: Attributes, Definitions and Formalisms," *Journal of Grid Computing*, Vol. 1, pp. 9-23, 2003.

[13] A. Mayer, et.al, "Meaning and Behavior in Grid-oriented Components," *Proceedings of Grid 2000, LNCS 2536*, pp. 100-111, Springer-Verlag, Heidelberg, 2002.

[14] D. Karger, et.al., "Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide." *Proceedings of the twenty-ninth annual ACM symposium on Theory of Computing* 1997, El Paso, Texas, United States *May 04 - 06, 1997,pp-654-663.*

[15] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network" *Proceedings ACM Sigcomm*, San Diego, CA, August 2001, pp. 161-172.

[16] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications" *Proceedings ACM Sigcomm*, San Diego, CA, August 2001, pp. 149-160.

[17] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems", *Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms* (Middleware 2001), November 2001, pp. 329-350.

[18] I. Clarke, S. Miller, T. Wong, Oskar Sandberg and Brandon Wiley "Protecting Free Expression Online with Freenet" IEEE Internet Computing, January-February 2002, pp.40-49.
[19] I. Foster, C. Kesselman, J. Nick, and S. Tuecke,"Physiology of the Grid: An Open Services Grid Architecture for Distributed Systems Integration," http://www.globus.org/research/papers/ogsa.pdf.
[20] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," http://www.globus.org/research/papers/anatomy.pdf
[21] P. Reynolds and A. Vahdat, "Efficient Peer-to-Peer Keyword Searching." *Proceeding of Middleware* 2003 (Rio de Janeiro, Brazil, June 2003).
[22] O. Gnawali "A Keyword-Set Search System for Peer-to-Peer Networks" (Masters Thesis, EECE Department, MIT, June 2002).
[23] A. Kronfol "FASD: A Fault-tolerant, Adaptive, Scalable, Distributed Search Engine" (Honors Thesis, Princeton University, May 2002).
[24] C. Tang1, Z. Xu and M. Mahalingam "PeerSearch: Efficient Information Retrieval in Peer-to-Peer Networks" Technical Report HPL-2002-198 Internet Systems and Storage Laboratory HP Laboratories Palo Alto July 12th , 2002.
[25] R. Oldfield and D. Kotz. "Armada: A parallel file system for computational grids) *Proceedings of the First IEEE/ACM International Symposium on Cluster Computing and the Grid* (Brisbane, Australia, May 2001) pages 194-201.
[26] M. Harren, et.al., "Complex Queries in DHT-based Peer-to-Peer Networks", , *Proceedings of First International Workshop on Peer-to-Peer Systems (IPTPS)*, (March 2002, Cambridge, MA).
[27] W. S. Ng, B.C. Ooi, K. L. Tan, and A. Zhou. "PeerDB: A p2p-based system for distributed data sharing" *Proceedings of the 19th International Conference on Data Engineering* (Bangalore, India, March 2003).
[28]See for example, the Proceedings of the 5th International Conference on Coordination Models and Languages, Springer Verlag, LNCS, 2315, (York, UK, April 2002).
[29] S. Gribble, A.Halevy, Z. Ives, M. Rodrig, and D. Suciu. "What Can Peer-to-Peer Do for Databases, and Vice Versa?" *Proceedings of the Fourth International Workshop on the Web and Databases (WebDB '2001) (Valencia, Spain, May 2001).*