

GridForce: A Comprehensive Model for Improving the Technical Preparedness of our Workforce for the Grid

B. Ramamurthy
CSE Department
University at Buffalo, SUNY
Amherst, NY 14260
716-645-3180 (108)
bina@cse.buffalo.edu
<http://www.cse.buffalo.edu/~bina>

Abstract

*An enormous challenge when the Internet matured into a mainstream technology was meeting the information technology workforce needs in a competitive business environment. In anticipation of a similar scenario for upcoming grid technology we are in the process of implementing a comprehensive multi-tier NSF-supported adaptation of grid technology in education. The project addresses the above mentioned challenge at three important levels of our educational system: the undergraduate, the graduate and the industrial training. Our grid technology-based curriculum has been developed for a sequence of two new courses for senior level undergraduates. The same courses would be taught at the graduate level with emphasis on research. Additionally, seminars are planned for spreading grid awareness to the local businesses and industries by using domain-dependent grid applications. This paper presents the details of the model we call **GridForce** (**Grid For Research, Collaboration and Education**) and our experiences with its implementation, with the objective of improving the technical preparedness of the workforce for the grid.*

Categories and Subject Descriptors

C.2.4 [Distributed Systems]: Distributed Applications,
K.3.2 [Computer and Information Science Education]
Computer Science Education, Curriculum.

General Terms

Distributed programming.

Keywords

CS education, design, laboratory experiments.

1. Introduction

The primary goal of *GridForce* is to promote grid awareness and technical readiness among all levels of our workforce. We accomplish this by the following strategies:

(i) Introducing grid computing in the CSE (Computer Science and Engineering) curriculum in the senior-level undergraduate and graduate courses. The courses CSE4/586 Distributed Systems and CSE 4/587 Information Structures are currently being offered as a two-semester course sequence in the CSE department of University at Buffalo.

(ii) Building laboratory prototypes that will support grid application development in the courses noted above. We are currently developing two grid prototypes, one with newer Dell Blades and another with old Sparc4 machines.

(iii) Conducting workshops for strengthening local industry workforce. We plan to offer our first seminar through the Center for Industrial Effectiveness (TCIE) at the University at Buffalo in March 2004.

(iv) Assessing the outcome and making ongoing adjustments. An external consultant is currently assisting in assembling a model of the courses and in carrying out a formal assessment of the effectiveness of the model.

This paper describes the educational model defined and implemented by GridForce. Related grid information and the current status of grid education are discussed in Section 2. Details of the various tiers of the multi-tier GridForce model, its implementation, and our experiences are detailed in Section 3. Available resources for adoption of various GridForce components and outcome assessment details are in Section 4. A summary of significant GridForce

contributions and acknowledgements are found at the end.

2. Project Background

A grid is a network of computational units cooperating to share compute cycles, data and other resources across multiple administrative domains, using an open and standardized service-based framework [8, 24]. Under NSF's Partnership for Advanced Computational Infrastructure (PACI) program [15], the scientific community is in the process of developing a national tera-scale infrastructure for high-performance computing. Prominent industries have eagerly embraced grid computing and are promoting it under different names such as utility computing and on-demand computing [11, 12, 13, 20]. Scientists as well as the practitioners believe that the grid developed for scientific computing is on the brink of making computing freely available as yet another "utility," similar in ease of accessibility to the power grid that supplies electricity and the telephone grid that enables voice communication.

We examined three universities that play a prominent role in shaping the grid technology: the University of Chicago, the University of California at San Diego (UCSD), and the University of Tennessee at Knoxville. The grid is presented as one of the topics in parallel computing courses [1, 2, 7], or as a seminar course [3]. The courses at UCSD taught by Dr. Fran Berman's research group, CSC160 (Parallel Computation) [2] and CSE225 (High Performance Computing and Computational Grids) [1] focused on high performance computing. We also studied the past offerings of cluster computing courses at the University at Melbourne which focused on parallel implementation of mathematical problems [4]. At the University of Wisconsin, the home of the Condor cluster/grid computing, the list of course offerings did not reflect any undergraduate or graduate courses in grid computing. We also observed that many schools have recently added programs and courses devoted to the emerging field of bioinformatics while they do not even have a single course devoted to grid computing. It is possible that many other schools may offer courses related to the grid but these have not been accessible due to the lack of a publicizing forum. Our GridForce is a comprehensive suite of courses and short courses rather than an entire program. Our experience is that this model is easier to sell to university administrators and can be implemented as a whole or in parts.

3. The GridForce Project

The GridForce project comprises three major components as depicted in the Figure 1. The **courses** (Section 3.1) play a central role with **laboratory infrastructure** (Section 3.2) and **research** (Section 3.3) components providing practical support. We will discuss the courses, laboratory projects and educational aspects in detail. Ongoing research projects will serve as topics for additional paper and are beyond the scope of this paper, the reason for indicating research subtree with no branches in the Figure 1. The project duration is for 2 years starting from the Fall 2003 that will allow for scheduling two offerings of each course.

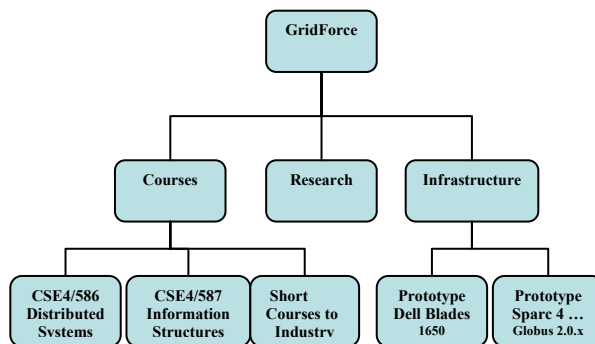


Figure 1: An Organizational Model of GridForce

3.1 Courses

The courses offered under this project form a virtual tiered structure with at least two fundamental tiers as described below. CSE4/586 is a course (undergraduate/graduate pair) in Distributed Systems. CSE4/587 (Information Structures) is a sequel to the Distributed Systems course. Focus of CSE4/586 is on fundamentals of grid computing whereas CSE4/587 deals with application development. Both courses include hands-on laboratory projects. Together these two courses form the Grid Services Developer Tier and are meant for the designers and developers of grid services and components. The next tier is an Industrial Training Tier and is meant for training industrial workforce in grid technology. It combines the salient features of academic courses with customized exercises and domain-dependent applications of interest to the workforce getting trained. Educators can extend this model by adding other tiers as seen appropriate.

3.1.1 Grid Services Developer Tier

Each course has a prescribed text book supplemented by recommended online literature. For

the CSE4/586 we used a traditional distributed systems text by Coulouris et al. [6] as the main text. The first half of the course covered the fundamentals of distributed systems and the second half explored grid fundamentals, grid programming, and grid application model. The highlight of the course is a set of grid programming labs designed to provide hands-on experience for the students. The objectives, course outline, weekly schedule, project descriptions and the all lecture material by date for the current semester can be found at [16]. We briefly explain the lab projects below.

Courses are assumed to span a 14-week semester. The course format involves a lecture that meets 3 X 50 minutes every week and a formal lab session that meets 50 minutes weekly. Prerequisites for the grid courses are data structures, algorithms and programming language courses. The fundamental concepts relevant to the lab and the lab description will be introduced and discussed during the lecture session and will be followed up during the lab session. Lab sessions will also cover details of configuration, packaging and deployment of the applications. Students will have open lab hours and help sessions where technical and debugging help for the projects will be available. Table 1 displays the lab exercises for the first (fundamental) course. The title of the labs and the learning objective are shown in this table. Students will be shown demos on the topic during the regular lecture sessions. Each student will complete an evaluation questionnaire that has both topic-related and learning-related questions. Students will submit online their solutions by a specified date.

Lab1 deals with Webservices [22] which moved the Internet from an information-delivery instrument to a computation-delivering channel. Introduction of the simple request and response model and standardization through SOAP (Simple Object Access Protocol) [22] help students understand (i) the “service” concept in its simplest form and (ii) at the same time realize the need for more sophisticated features such as lifecycle management and notification [19]. We use a simplified version of the Webservices tutorial provided at the Java Webservices site [25] to implement a version of a weather services application. The projects also use the Oracle 9i relational database for persistence, and Ant tool from Apache for building the code. Content-wise the lab also deals with the fundamental distributed systems issues of service registry, discovery and lookup.

Exercise	Topic	Learning Objective
Lab1	Webservices	To understand the alignment of the grid technology to Web Services Definition Language (WSDL) and service description using WSDL.
Lab2	Grid Infrastructure	A simple Java-based grid framework based on [14].
Lab3	Grid Programming	Design and implement a grid-based service using Globus 3.0.2

Table 1 Suggested Lab Projects for CSE4/586 Distributed Systems Course

Lab2 Currently many toolkits such as Globus Toolkit 3.0.2 [10] and Condor 6.5.5 [5] are available to implement the grid framework. However these frameworks are production-quality and are quite complex for the students to understand, deploy and take apart to study and experiment with the code. Lab 2 deals with building a minimal grid framework based on the article “A do-it-yourself framework for grid computing” by Anthony Karre in Java World [14]. Our focus in this project is on the client-side of the grid computing. The framework given in the paper offers these features: (i) machine independence through Java, Apache Tomcat servlet container and Apache Axis SOAP implementation, (ii) Security and scalability achieved through the use of SOAP-based Web services for client-server communication and (iii) task abstraction achieved through the use of jar files, and Java classloader.

In the current assignment of Lab2 we use *SOAP with Attachments API for Java* (SAAJ) [22] instead of the Apache Axis specified in the Karre’s paper so as to work at a lower level of abstraction. Students also build a custom classloader and a simple user interface to suit the service (weather service) that has been implemented.

Lab3 deals with implementation of a grid service and an application that uses the grid service. The grid software used for this lab is Globus 3.0.2 core [19]. Students study in detail the Open Grid Services Architecture (OGSA) [9, 23] and Open Grid Services Infrastructure (OGSI) [24]. A comprehensive tutorial on Globus Toolkit (GT3) at [21] and the GT3 core are discussed during the lecture sessions. The students are required to build a many versions of the weather (grid) service from a basic version to a sophisticated one with

features listed in the Globus core distribution. The software that we will be using is the core of the Globus Toolkit 3.0.2. The core of Globus can be downloaded from [19]. Details of the core are available in a white paper on the core services at [19]. This white paper also contains a javadoc-style Grid Services API description, User's Manual and a Programmer's Manual. The user's manual provides the instructions to compile, build, convert, deploy and test a grid service. The programmer's manual provides the details of writing a grid service, the various programming choices available, and deployment description. A samples directory in the core package provides a numerous examples illustrating the various grid services features. Each student installs the Globus core in a special project space allocated for the course and develops and deploys a service on the server. The service is tested using a simple user application.

Projects in the second course CSE4/587 involve applying grid technology to solve problems in specific application domains as shown in Table 2. We have chosen two specific areas of topical interest to grid technologists. The first lab deals with a scientific application in bioinformatics. Lab2 is based on the commercial domain. In this lab we plan to study something topical such as volatility in the stock market and the models for it. Lab3 will be designing and/or modifying a grid-level service such as security and QoS and also defining a business process using a complex grid service. Students will have to come up with original ideas in this lab. There are plans to introduce the Java 2 Enterprise Edition (J2EE) framework for the early projects to allow the students to do a comparative study of the technologies. The author strongly believes the two technologies J2EE and Grid will have to unify to complement and strengthen each other as standard computational framework of the future.

3.1.2 Industrial Training Tier

For this tier, two approaches are possible: (i) an executive summary or a business overview for strategic decision makers or business people and (ii) a hardcore developer point of view. For business people we plan to offer a two-hour breakfast seminar (jointly sponsored by NSF, CSE Department and our University's Industrial Liaison office). Presentations for the developers will be characterized by deeper coverage in a selected area, faster pace and customized mode of delivery. The author of this paper conducts regular training sessions in object-oriented design, programming, and similar topics to the local industry. She has also conducted a how-to of industrial training to educators at national conferences [17, 18]. The

author feels that industrial training (say, a six week, 3 hours per week, lab included) focused on a specific topic is one of the best ways to retrain the existing workforce. The industrial training will serve a dual purpose: (i) retraining the IT workforce to be ready for the compute grid, and (ii) field-test the grid technologies for practicality and usability in their respective application environment.

Exercise	Topic	Learning Objective
Lab1	High performance Scientific Application in bioinformatics.	Study requirements of scientific domain and implement. Ex: micro-array analysis
Lab2	Commercial Application	Study requirements of commercial domain and implement. Ex: Stock Market
Lab3	Defining a high-level grid service	Ex: Workflow service, a business process, improvements to QoS

Table 2 Suggested Lab Projects for CSE4/587 Information Structures Course

3.2 Infrastructure

Another important component of the educational model is the laboratory infrastructure. We are currently building two different experimental research and development grids: (i) 40 Sun Microsystem's Sparc 4 discarded computers (originally used for graduate students desktops) with an Ultra Sparc 5 as front-end gatekeeper all running Solaris 8.0 operating system and the Ultra Sparc running Globus 2.0 grid software, and (ii) a grid with four newer Dell blades 1650 hardware, a combination of FreeBSD and Red Hat Linux 9.0 operating systems, and all running Globus Toolkit 3.0.2. Only the computational resources are identified here. We are at a very early stage of addressing storage needs. Additionally both infrastructures are currently undergoing tests for full-scale deployment for student use in the courses for the Spring 2004.

3.2.2 Prototype 1: Using old Sparc 4 Machines

The goal of the infrastructure is to run remote job submissions in a distributed manner on a Sun Microsystems computational cluster running Globus. The grid is primarily composed of 40 Sun Sparc

machines, which form computational nodes, headed by a front-end Sun server running Globus. The internal Class C network is set up using custom NFS, NIS and jumpstart servers. The jumpstart server is an operating system server providing remote Solaris 8 installation for clients over the network. The installation scripts are custom-written facilitating running of jobs in a distributed manner.

The network also has an NFS service running that provides remote file mounts and access. The server exports its *home* and *util* directories for NFS clients in the network. Name service and network information is provided using an NIS+ service running along with the NFS server. The central server runs a custom version of Globus binaries handling remote Job submissions. The custom Globus binary package installed is comprised of binaries of 2.x versions optimized for specific performance issues. The grid certification mechanism is a DOE certification process providing host and user certificates.

3.2.2 Prototype 2: Using Dell blades 1650

This experimental grid is set up as one utility server and three compute nodes. FreeBSD was chosen for the utility server. This server is designed to provide network gateway/firewall services as well as basic UNIX-level account authentication (NIS) and file (NFS) services. Some of the support services for Globus also run on this server. The three compute nodes are running RedHat-9 and Globus Toolkit version 3 (GT3).

3.3 Research

The main focus of our group is applied research that can potentially expand the grid technology to support mainstream applications. Students in the courses described above each worked on a poster that explained a possible application of the grid. Some of the ideas include: (i) agent-based grid security application, (ii) grid application to pick stocks, (iii) income tax return filer grid service based on high-trust of the grid technology and (iv) grid application development environment. We plan to implement these projects as labs for the second course CSE4/587. The details of these projects are topics for future publications from our group.

4. Resources for Adopters

All the material needed for adoption and adaptation of GridForce courses is available on our webpage: www.cse.buffalo.edu/gridforce. An important component of the course model is the outcome

assessment process. We have an extensive outcome assessment questionnaire prepared with the help of professional evaluators. For example, an assessment questionnaire for the course CSE4/586 has about 42 multiple choice questions and 4 short answer questions comprehensively covering all the elements of the course model. We will make this and similar forms available for educators to reuse. The details of the infrastructure will be made available as soon the testing is completed.

5. Summary

We have presented a comprehensive model addressing the need to improve awareness and technical preparedness of our workforce. The outcome of this model will impact a wide variety of audiences from undergraduate students to business strategists. All the information related to the model is web-accessible to potential adopters. We have plans to apply our experience to teaching fundamental concepts related to grid and to develop grid-based curriculum for computer architecture courses.

6. Acknowledgements

This work is partially supported by NSF grant CCLI A&I DUE 0311473.

The author acknowledges Ken Smith and Karthikram Venkataramani for help with the Dell and Sparc infrastructures respectively. We thank Center for Computational Research (CCR) at University at Buffalo for the grid expertise they have provided and for hosting the Sparc infrastructure. The author would like to acknowledge the anonymous reviewers for their feedback on the content of paper, and Dr. S. Goldberg for proof reading the paper.

7. References

- [1] F. Berman, CSC225 High Performance Distributed Computing and Computational Grids, <http://www-cse.ucsd.edu/classes/sp00/cse225/>, Spring 2000
- [2] F. Berman, CSC160 Parallel Computation, <http://juggler.ucsd.edu/~nadya/160/> Fall 2000.
- [3] H. Bos, Seminar on Grid Computing, University of Amsterdam, Netherlands, <http://www.liacs.nl/~herbertb/courses/grid/>, 2001.
- [4] R. Buyya, 433-498: Cluster and Grid Computing, <http://www.cs.mu.oz.au/~raj/grids/course>, 2002.
- [5] Condor High Throughput Computing, <http://www.cs.wisc.edu/condor/>

- [6] G. Coulouris, J. Dillmore, and T. Kindberg. Distributed Systems: Concepts and Design, Third Edition, Addison-Wesley Publication, 2001.
- [7] J. Dongarra, CSE594: Understanding Parallel Architectures: From Theory to Practice, University of Tennessee at Knoxville, Spring 2002.
- [8] I. Foster, C. Kesselman, S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International J. Supercomputer Applications*, 15(3), 2001, See <http://www.globus.org/research/papers/anatomy.pdf>
- [9] I. Foster, C. Kesselman, J. Nick, S. Tuecke. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002. See <http://www.globus.org/research/papers/ogsa.pdf>
- [10] Globus Toolkit 3.0: Open Grid Services Architecture, <http://www.globus.org/toolkit/>
- [11] Grid Computing at IBM, <http://www-1.ibm.com/grid/>
- [12] Grid Computing: Making the Global Infrastructure a Reality. Edited by F. Berman, G.C. Fox, A.J.G. Hey, Wiley and Sons, 2003. ISBN: 0-470-85319-0
- [13] Information Power Grid (IPG) is NASA's high performance computational grid. <http://www.ipg.nasa.gov/>, Nov. 2002
- [14] A. Karre. A Do-it Yourself Framework for Grid Computing. <http://www.javaworld.com/javaworld/jw-04-2003/jw-0425-grid.html>, Java World, April 2003.
- [15] NSF: NSF Announces Continuing Steps to Enhance Cyberinfrastructure, <http://www.cise.nsf.gov/news/cybr/cybr2.cfm> October 2003.
- [16] B. Ramamurthy. GridForce: Grid for Research, Collaboration and Education. <http://www.cse.buffalo.edu/gridforce/index.htm>, 2003.
- [17] B. Ramamurthy. A Multi-tier Adaptation of Grid Computing in Computer Science Curriculum, to be presented at SIGCSE 2004 poster session, Norfolk, VA, 2004.
- [18] B. Ramamurthy. Industrial Training Know-how for Educators. Workshop presented at ACM SIGCSE 2000, Austin, TX.
- [19] T. Sandholm and J. Gawor. Globus Toolkit 3 Core – A grid Service Container Framework, Globus Project, July 2003, http://www-unix.globus.org/toolkit/3.0/ogsa/docs/gt3_core.pdf
- [20] P. Shread, Sun Sees Campus Grids as next Stage, Grid Computing Planet on-line magazine, January 2002. See http://www.gridcomputingplanet.com/news/article/0..3281_952091.00.html.
- [21] B. Sotomayor. The Globus Toolkit3 Programmer's Tutorial, 2003. <http://www.casa-sotomayor.net/gt3-tutorial/>
- [22] K. Topley. Java Web Services in a Nutshell, O'Reilly Publishers, June 2003.
- [23] Towards Globus Toolkit 3.0: Open Grid Services Architecture, <http://www.globus.org/ogsa/>.
- [24] S. Tuecke, K. Czajkowski, I. Foster, J. Frey, S. Graham, C. Kesselman, T. Maguire, T. Sandholm, P. Vanderbilt, D. Snelling, "Open Grid Services Infrastructure (OGSI) Version 1.0."; Global Grid Forum Draft Recommendation, 6/27/2003.
- [25] Web Services Tutorial, Sun Micro Systems Inc., <http://java.sun.com/webservices/tutorial.html>, 2003.