

An Effective Model for Computer System Building Projects in Computer Engineering and Computer Science

Yul Chu and Jin Hwan Park

Abstract—E-learning has become an important part in education in this digital era. We developed an effective eLearning model for system building projects in the disciplines of computer engineering and computer science, and applied it to a project of building a high performance computer system, Beowulf cluster. The aim of the practice is to elevate the level of students' knowledge and technology to build a clustered computer system by facilitating on-line resources, including related research outcomes and open-source software. The project is designed for team working and students in a team collaborate to complete both hardware setup and open-source software installations based on the manuals and guides accessed from Internet sources. The project was actually conducted in Parallel Computing course in UTPA and Parallel Processing course in CSU, Fresno, and the resulting systems were tested with a number of benchmark parallel programs for performance. Based on the successful outcome of our students, we believe that the developed eLearning model is highly effective for system building projects in the disciplines of computer engineering and computer science.

Index Terms—Cluster, e-learning, Internet, parallel computing, high performance system.

I. INTRODUCTION

With the rapid development of the Internet technology, eLearning approach has been a new trend in the pedagogy. Since traditional classroom lecture based methods of transmitting an instructor's knowledge to students have a lot of limitations in their educational capacity eLearning based methodologies have rapidly been adapted in a variety of disciplines, including liberal arts, science and engineering.

In this paper, we propose an efficient eLearning model, which is developed for manipulating computer system building projects in the disciplines of computer engineering and computer science, and show its adaptability on a high performance computer system building project [1]. In the disciplines of computer engineering and computer science, course projects have been used as important tools for making students understand course materials effectively and apply the theoretical knowledge to solve real world problems effectively [2]. Recently, Internet sources have become important parts of conducting such projects since they provide open source technologies and software [3]. However, it is difficult to find a proper guide for using Internet oriented technologies and materials to promote students' ability of

completing the course projects in the disciplines of computer engineering and computer science. This becomes the motivation of our efforts of developing an efficient e-learning model, which elevates students' ability to utilize the Internet oriented resources to complete computer system building projects. The application project conducted in our practice is building a high performance computer system, Beowulf cluster, using multiple PCs or workstations [4], and students are grouped in teams and cooperate with team members to complete the project. Students are open to use any technologies and prior research outcomes collected from Internet sources, as well as using open source software. The resulting product is a clustered computer system, which is a type of popularly used high performance computer systems for scientific computations demanded in the fields of computational biology, chemistry, physics, etc.

The rest of this paper is organized as follows. Section II describes the proposed eLearning model for computer system building projects. Section III describes the application course project conducted. Section IV describes the evaluation of the project and students' feedback, and Section V concludes the paper.

II. PROPOSED E-LEARNING MODEL FOR COMPUTER SYSTEM

E-learning is defined as "the use of the Internet to improve the quality of learning by facilitating access to resources and services anytime and anywhere [5]." Therefore, the eLearning approach is quite different from conventional classroom based learning approaches since it is efficient to elevate the level of students' knowledge through students' dynamic participation in collecting information from Internet sources. The benefits of eLearning are as follows [6]: 1) Distant learning (anywhere and anytime), 2) A blended learning/teaching approach (classroom based and eLearning) is available, and 3) The use of modern technology (web based technology, multimedia, etc.).

In this section, we describe our proposed eLearning model, which is developed based on the Schulman's 6-stage learning process [7], i.e., adapting Internet oriented technologies into the Schulman's model. Schulman [7] proposed the following 6-stage learning process for learners: 1) engagement and motivation, 2) knowledge and understanding, 3) performance and action, 4) reflection and critique, 5) judgment and design, and 6) commitment and identification. We reflect the six stages of the Schulman's learning process to design appropriate eLearning methodologies for computer system building projects.

Fig. 1 shows the flowchart of our proposed 6-stage e-learning model for computer system building projects. Detailed descriptions of the stages are as follows.

Manuscript received March 11, 2013; revised May 19, 2014.

Yul Chu is with Electrical and Computer Engineering, University of Texas-Pan American, Edinburg, TX, U.S.A. (e-mail: chuy@utpa.edu).

Jin Hwan Park is with Computer Science, California State University, Fresno, CA, U.S.A. (e-mail: jpark@csufresno.edu).

- 1) Project introduction (classroom based learning): Instructor defines the topic of the course project and explains educational contents, such as learning objectives, concepts, materials, resources, constraints, and expected output;
- 2) Task analysis (classroom based learning): Instructor hands out the class project guideline, which includes detailed instructions of scheduled tasks including proposal submission, design and implementation, final report writing and demonstration;
- 3) Preliminary research (e-learning stage): Students (individual or team) are asked to collect all needed information and materials to complete the project from Internet sources and submit a proposal for the implementation methodology to the instructor. If the proposal is rejected, students review the tasks and resubmit the proposal until accepted;
- 4) Design (e-learning stage): Students (individual or team) configure the resulting computer system based on the technologies and materials collected from the Internet sources. This includes both hardware configuration and open-source software installation plan according to the project objectives;
- 5) Implementation (e-learning stage): Students (individual or team) implement the system based on their design completed in Stage 4 and technologies gained from Internet sources. If problems are found, students revisit the design, correct the problems and redo the implementation work until completed;
- 6) Evaluation (e-learning stage): Students (individual or team) evaluate the performance of the resulting computer system using benchmark programs accessed from Internet sources and submit the final report to the instructor, as well as doing demonstration.

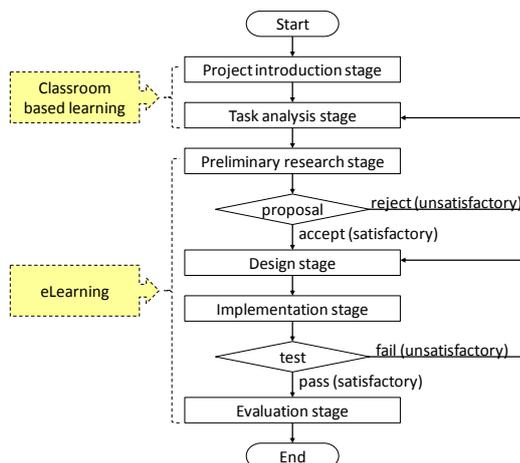


Fig. 1. Flowchart of the proposed e-learning model for computer system.

Our proposed eLearning model for computer system building projects is generic for both individual and team projects. As described above, the first two stages of the model are directly based on the concepts of the first two stages of the traditional Schulman's learning process [7], i.e., the first stage of the proposed model reflects the engagement and motivation, and the second stage reflects the knowledge and understanding. From the third stage of the proposed model, all stages are Internet based. In particular, the third stage does very important roll in our model since it is the first

phase of accessing Internet sources and affects the quality of the entire project work. With the features shown above, our proposed e-learning model for computer system building projects differs from traditional classroom based learning process, conventional e-learning and blended learning approach, which simply combines positive aspects of the traditional and conventional eLearning approaches [8].

III. PROJECT: BUILDING A BEOWULF CLUSTER SYSTEM

In this section, we describe how the proposed eLearning model is utilized to manipulate a course project of building a high performance computer system, a Beowulf cluster, and to accomplish the goal of educating students more effectively. The goal of the project is building a clustered computer system with both hardware set up and system software installations, and the project was conducted in Parallel Computing course at the University of Texas, Pan American and in Parallel Processing course in California State University, Fresno. For the purpose of keeping the consistency of describing the practice, we use the experiment conducted in UTPA in this paper.

Clustered computer systems have been used for high performance and parallel computations in a variety of application areas [2]-[4]. Therefore, the implemented system should be tested with real world's application programs for its performance. All the information and materials for design, implementation, testing are accessed from Internet sources through eLearning approaches. The project is designed for team working, i.e., 3 students in a team, and each team is required to build a simple clustered system, known as Beowulf cluster [4], using any number/type of PCs or workstations available in the lab. For example, a basic Beowulf cluster system can be built from only two PCs (a master node and a computing node) and acts as a high performance computer system. This single image concept is realized from installing a package of system software, such as Rocks Cluster Distribution [9], which is open-source software downloadable via the Internet. A clustered system allows multiple nodes (PCs or workstations) work in parallel with multiple or partitioned code/data and the results are assembled in the master node. The goal of the project is to let students promote their practical knowledge level and research experience through eLearning with the Internet technology. In each stage of the proposed eLearning model, the following activities are done for the project.

A. Project Introduction and Task Analysis (1st and 2nd Stages)

In the Project introduction stage, the instructor defines the topic of the project, i.e., building a clustered computer system, and prepares related materials, such as slides for explaining the project. Students are informed that the cluster system is a type of high performance computer system and used in many application areas with its parallel computing power. Students are also informed that the resulting cluster system consists of hardware components (multiple PCs or workstations and a network switch), Linux operating system, networking software (e.g., DHCP and NFS), queuing system software (e.g., Torque and Maui), and monitoring system software

(e.g., Ganglia), as well as other system software including OpenMP, OpenMPI and MPICH. Course materials prepared for the project are divided into two groups, one is for hardware and the other is for open-source software, and are presented in class lecture sessions.

In the following Task analysis stage, more detailed guides for completing the project are given to students. Since the project is designed for team working, students are asked to form teams with 3 members each and elect team leaders. To achieve the goal of completing the project, each team clarifies all the issues based on the project guideline, distributes tasks to team members, and sets up the schedule to complete the project. The major tasks to complete are: 1) preliminary research and proposal submission, 2) hardware structural design and implementation, 3) open-source software access and installation, 4) performance evaluation and demonstration, and 5) writing project report.

B. Preliminary Research (3rd Stage)

In this stage, each team works for collecting all needed information and materials from the Internet sources. This includes the required knowledge of building the cluster system, detailed methodologies of both hardware setup and software installation, open-source software codes, etc. From this preliminary research, each team is required to submit a proposal based on the Internet oriented study. The proposal should include specific configurations how to design, implement and evaluate the cluster system. Based on the instructor’s feedback on the proposal, unsatisfactory proposals are required to be resubmitted until passed.

C. Design (4th Stage)

After submitting the proposal and the proposal is approved by the instructor, each team works for designing its own cluster system by configuring hardware structure, component connection methodologies, software installation plan, etc. The followings are sample hardware/software configurations done by a selected team, say Team1.

1) Hardware configuration

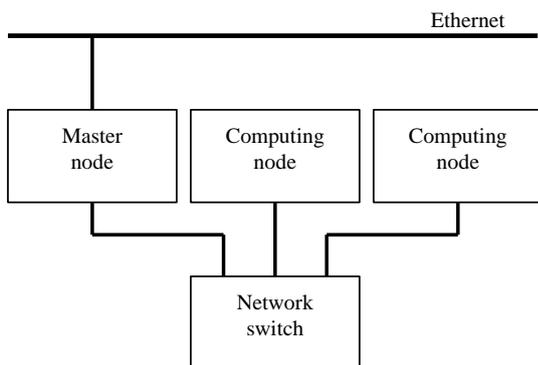


Fig. 2. Conceptual view of Beowulf cluster with 3 nodes.

Team1 determined the cluster system with three DELL Optiplex 745 workstations (one master node and two computing nodes) and a 100 Mbps Ethernet switch. The master node has a 1.8 GHz Intel Core 2 processor and each compute node has a 1.8 GHz Intel Core 2 processor. Each node in the cluster system is equipped with 2GB memory and a 149 GB hard disk. In order to connect the nodes of the cluster system and connect the cluster system to the Internet, one Ethernet switch, Netgear FS608, is used. This component

allows the system to be updated through the Internet and makes SSH access to the system available. The two computing nodes are connected to a shared keyboard, a monitor, and a mouse with a KVM switch to reduce the cost of implementation. Fig. 2 shows the conceptual view of the hardware connection for the Beowulf cluster. Fig. 3 – Fig. 5 shows actually used hardware components, which are master node, two computing nodes and an Ethernet switch, respectively.

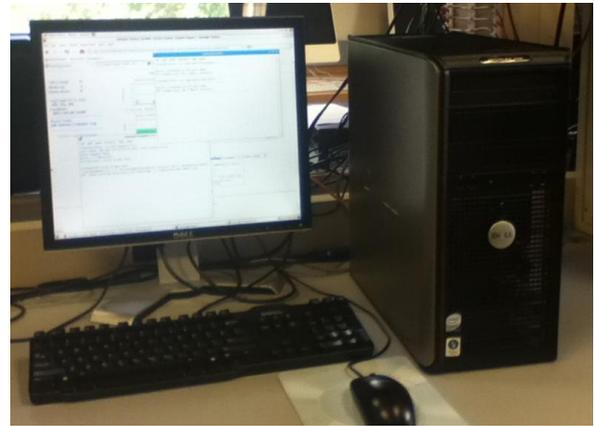


Fig. 3. Master node, DELL Optiplex 745 workstation.



Fig. 4. Computing nodes, DELL Optiplex 745 workstations.



Fig. 5. Netgear FS608 Ethernet switch.

2) Software configuration

TABLE I: OPEN-SOURCE SOFTWARE FOR BUILDING THE CLUSTER SYSTEM	
Name	Description
Area51	File and Kernel integrity analyzer
RedHat Enterprise Linux	Operating system
DRBL, SSH, NTP	Networking system
Ganglia	Cluster monitoring system
HPC	High performance computing tools [MPI, PVM, Benchmarks]
SGE	SUN Grid Engine Scheduler
Xen	Virtual Machine
Torque and Maui	Queuing system

Team1 configured needed software, which are open-source software downloadable from Internet sources, as shown in Table I. The operating system chosen for the cluster system is Rocks Cluster Distribution [9], which is based on the RedHat Enterprise Linux. Several pieces of software are

included in the Rocks Cluster Distribution package and they will be enabled during the installation process. These programs are to be installed in an appropriate order in the implementation stage.

D. Implementation (5th Stage)

In this stage, each team implements the design of the cluster system by setting up and connecting the configured hardware components and installing the configured software in an appropriate order. We describe below the implementation process done by the selected team (team1), which we mentioned in Section III-(C).

The first step is building the master node with the clustering operating system. To install the Rocks Cluster Distribution software package, students followed the steps listed in the online installation guide [10], [11]. First, a set of five required CDs were downloaded from Rocks website [9]. Second, the master node with the Kernel/Boot CD was booted and the *build* option was selected. Using the guided installer, students were able to configure the network for IPv4 and DHCP, download all the available software components from the CDs, select a hostname for the frontend. Lastly, defaults for the remainder of the network settings were accepted, a password for the root user was entered, the time zone was settled, and automatic disk partitioning was selected.

The installation of the computing nodes was relatively easier comparing to the process of building the master node. The first step was running *insert-ethers* command on the front-end node and selecting *compute* for appliance type. Second, the BIOS of each computing node were configured for network booting. After restarting the computing nodes, installation was processed automatically. Finally, students created user accounts by running *useradd* and *username* commands and synchronized the accounts throughout the cluster via *rocks sync users* command.

Implementation of the system also includes installing benchmark application programs to measure the performance of the resulting cluster system. In fact, the Rocks Cluster Distribution package contains some suggested benchmark application programs. However, students realized that LINPAC, which is a typical benchmark program for measuring the parallel computing power, was not included in the package and separately installed LINPAC by following the guides accessed from the Internet [12].

E. Evaluations (6th Stage)

Followed by the successful implementation of the cluster system, each team is required to do performance evaluation, do demonstration and write the final report. Performance measurement is done with popularly used benchmark programs. We describe here the performance measurement result from the selected team (team1).

Team1 chose to run three benchmark programs, which are LINPAC, IOzone, and Iperf, as suggested in the Rocks Introduction to Benchmarking [12]-[16]. LINPAC has been a popular benchmark program in the field of high performance computing. It measures the system's floating-point computing power on solving a matrix problem using LU decomposition [12]. IOzone measures the speed of the file system and Iperf measures the speed of the interconnection

network by assigning traffic load between two nodes in the cluster system [15]. This benchmark application is started by running the server process in a node and the client process in one other node. Fig. 6 shows the performance of the cluster system, which was observed by Team1, measured with LINPAC application.

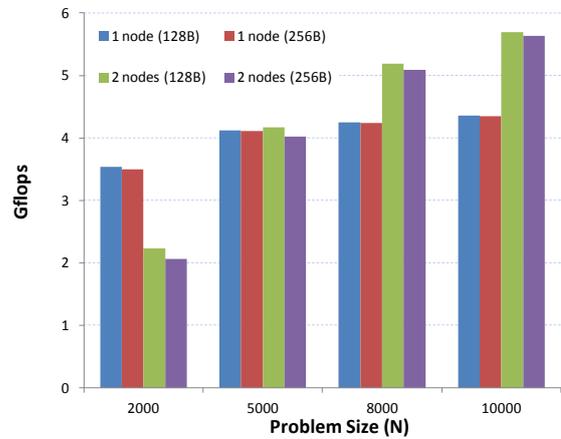


Fig. 6. Performance vs. problem size using 1 processor per node.

In the figure, y-axis represents performance in GFlops and x-axis represents the problem size, i.e., matrix size. As shown in the figure, the LINPAC application is executed with four problem sizes, i.e., $N=2000$, 5000, 8000 and 10000, and two block sizes, i.e., 128 bytes and 256 bytes. Tests for performance comparison were done with the cluster system with one node and two nodes (each node has one processor). Fig. 6 shows that the cluster system with one node achieved better performance than the system with two nodes for the small problem size of 2000 due to communication overhead. However, with the bigger problem size (over 5000), the system with two nodes showed much better performance than that of the system with one node.

In this final stage of the project, each team is required to submit the final report describing the work, results and thoughts, after doing a demonstration.

IV. PROJECT EVALUATION AND STUDENTS' FEEDBACK

The evaluation of the computer system building project is based on the correctness and efficiency of the resulting system. The following three fields are separately graded and combined for the final grade of the project: 1) Preliminary research results and proposal conducted in the 1st to 3rd stages of the proposed eLearning model, 2) System design and implementation quality conducted in the 4th and 5th stages, and 3) Performance measurement methodologies and the final report conducted in the 6th stage. In the case of a team project, an additional evaluation category is checking the level of cooperation within a team since the goal of the project is achieved effectively by sharing ideas and technologies among team members.

In our practice at UTPA, students were grouped in 4 teams. Each team's project work was graded for the above three fields, as well as the level of the team cooperation since the project was a team project. Scales used are preliminary study and proposal quality (20%), correctness and efficiency of design and implementation (50%), testing and performance

evaluation (20%) and the team cooperation level (10%). As a result, three teams from four (75%) got grade A (excellent, over 90%), and one team (25%) received grade B (good, between 80% and 90%). In CSU, Fresno, only one team (with two members) selected the project of building the cluster system since the course project allowed students to select a topic from multiple choices, and the project was graded based on the same criteria and the team got grade A too. Students' feedbacks on the project work were very positive and we believe that the students were strongly stimulated by the eLearning methodology used for the project comparing to the classical classroom based approaches.

Based on this practice, we could say that the computer system building project manipulation based on the proposed eLearning model is pretty successful since all the students clearly understood the concept of building the cluster system, gained knowledge and technology through accessing appropriate Internet sources, and could implement the resulting system from their knowledge and technology gained through the eLearning.

V. CONCLUSION

An efficient eLearning model for computer system building projects in the disciplines of computer engineering and computer science is developed and applied to a course project. The application project in our practice is building a Beowulf cluster system, which is a high performance computing system for scientific applications. With the traditional classroom based learning, it has been difficult to conduct such project work in these high tech fields of disciplines. In fact, typical projects have depended on simulations using given simulation programs or by developing such simulation programs in a fixed learning environment. With the proposed eLearning model we could conduct the project in a dynamic learning environment, which involves utilizing the pool of knowledge and technology available through the Internet sources.

The developed eLearning model is generic for a wide range of computer system building projects and deals with both individual and team projects. Our experimental results demonstrated that the developed eLearning model is efficient for elevating the level of students' knowledge and technology to complete the project work in a team working environment. In our practice conducted in two different campuses, four teams from five performed excellent jobs and the remaining one team also achieved the goal of the project with good standing. Based on our prior experience of conducting general course projects with fixed learning environment, this ratio of succeeding on the project is considerably high. We believe that the open and dynamic learning environment with the proposed eLearning model greatly contributes to the students' performance on the

system building projects with strong stimulations and improves the quality of learning.

REFERENCES

- [1] D. A. Patterson and J. L. Hennessy, *Computer organization and design: the hardware/software interface*, fourth edition, San Francisco, California: Morgan-Kaufmann, 2012.
- [2] V. P. Heuring and H. F. Jordan, *Computer Systems Design and Architecture, second edition*, Upper saddle river, New Jersey: Prentice Hall, 2004.
- [3] M. Rosenberg, *E-Learning: Strategies for Delivering Knowledge in the Digital age*, McGraw-Hill Education, 2001.
- [4] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*, fifth edition, San Francisco, California: Morgan-Kaufmann, 2012.
- [5] EC, *Communication from the commission to the council and the European parliament: the e-learning action plan*, Brussels, vol. 28, no. 3, 2001.
- [6] JISC Digital Media. (April 20, 2013). [Online]. Available: <http://www.jiscdigitalmedia.ac.uk>.
- [7] L. S. Schulman, "Making differences: A table of learning, in change," vol. 34, no. 6, pp. 36-44, November/December 2002.
- [8] Blended learning. (April 20, 2013). [Online]. Available: <http://www.nwlink.com/~donclark/hrd/elearning/blended.html>
- [9] Rocks Clusters. (March 11, 2011). [Online]. Available: <http://www.rocksclusters.org/wordpress>.
- [10] 2.2 Install and Configure Your Frontend: RocksClusters. (March 11, 2011). [Online]. Available: <http://www.rocksclusters.org/roll-documentation/base/5.4/install-frontend.html>.
- [11] 2.3 Install Your Compute Nodes: RocksClusters. (March 11, 2011). [Online]. Available: <http://www.rocksclusters.org/roll-documentation/base/5.4/install-compute-nodes.html>.
- [12] Z. Z. Chen, J. Dongarra, P. Luszczek, and K. Roche, "LAPACK for clusters project: An example of self adapting numerical software," *Hawaii International Conference on System Sciences HICSS-37*, Hilton Waikoloa Village, Big Island, Hawaii, January 5-8, 2004.
- [13] Introduction to Benchmarking: RocksClusters. (April 19, 2011). [Online]. Available: <http://www.rocksclusters.org/rocksapalooza/2006/lab-benchmarking.pdf>.
- [14] C.-H. Hsu and U. Kremer, "IPERF: A framework for automatic construction of performance prediction models," *Workshop on Profile and Feedback-Directed Compilation (PFDC'98)*, Paris, France, October 1998.
- [15] Iozone. (April 20, 2011). Iozone File System Benchmark. [Online]. Available: <http://www.iozone.org>.
- [16] Mohammad. (April 13, 2011). LINPAC, How to install! *Online Posting*. [Online]. Available: <http://forum.senmurv.org/topic/linpack-how-to-install#post-53>.

Yul Chu is an assistant professor in the Department of Electrical Engineering at University of Texas Pan American. He received his Ph.D. in electrical and computer engineering from University of British Columbia, Canada in 2001 and MS in electrical engineering from Washington State University in 1995. His current research interests lie in the area of low-power embedded systems, high performance computing, parallel processing, cluster and high-available architectures, computer networking, digital system design, etc.

Jin H. Park is an assistant professor in the Department of Computer Science at California State University, Fresno. He received his Ph.D. degree in computer science from Oklahoma State University in 1998. His research interests include high-performance computing, parallel and distributed processing, bioinformatics and computational biology, and reconfigurable computing.