A Computer-supported Collaborative Learning Platform
Based on Clouds

Jiandun LI1, Junjie PENG1,2,†, Wu ZHANG1, Fangfang HAN1, Qin YUAN1

1School of Computer Engineering & Science, Shanghai University, Shanghai 200072, China
2Joint Lab of Cloud Computing, High Performance Computing Center, Shanghai University, Shanghai 200072, China
Key Laboratory of Computer System and Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China

Abstract

Computer-supported collaborative learning (CSCL) is an emerging branch of learning science concerned with studying how people can learn together with the help of computers. As an indispensable ingredient, computer mediation evolves through couples of phases, e.g. centralized server, peer-to-peer network, grid computing. Nevertheless, with daily rising trend on requirement’s dynamic changes in service, existing models fail short to respond on demand. Thus, in this paper, by taking advantage of cloud computing, we propose a feasible CSCL platform and the experiments show that it can not only fulfill the basic requirement of CSCL, but also respond to learner’s dynamic need on demand.

Keywords: Computer-supported Collaborative Learning (CSCL); Cloud Computing; Virtual Network

1. Introduction

The term computer-supported collaborative learning (CSCL) was used as early as 1989 by O’Malley and Scanlon and was recognized by Koschmann as an important area of research focus in 1996 [1]. In recent years, CSCL is emerging as a dynamic, interdisciplinary and international field of research focused on how technology can facilitate the sharing and creation of knowledge and expertise through peer interaction and group learning processes [2]. It is at least an inclusion of collaboration, computer mediation and distance education, yet the interplay of learning with technology turns out to be quite intricate [3]. By now, through taking advantage of couples of computer technologies, e.g. centralized server, peer-to-peer network, grid computing, CSCL gains wide acceptance from diverse learners. However, with daily rising trend on requirement’s dynamic changes in service, exiting models fall short to respond on demand.

It is said that “future CSCL studies should focus less attention on the question of whether CSCL is better than face-to-face collaborative learning, but rather focus on what is uniquely feasible with new technology”[2]. As a new promising paradigm, cloud computing can offer utility-oriented IT services to users based on a pay-as-you-go model. In Berkeley’s View [4], “cloud computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even

† Corresponding author.
Email addresses: jjie.peng@shu.edu.cn (Junjie PENG).

1553-9105/ Copyright © 2011 Binary Information Press
November, 2011
more attractive as a service and shaping the way IT hardware is designed and purchased”. It can also make good use of economies of scale, and dynamically deliver/configure almost any IT related services on demand [5]. Moreover, it can conserve more energy, which is the ideal computing platform for the coming low-carbon economy [6-8].

Nevertheless, little work has been done to tackle collaborative learning in particular onto cloud platforms as yet. Thus, it remains to be a challenging problem to make good use of the current IT technology for CSCL. In this paper, we propose a feasible platform based on clouds and the experiments show that it can not only fulfill the basic requirement of CSCL, but also respond to learner’s dynamic need on demand.

The remainder of this paper is organized as follows. Section 2 describes some studies related to our research; Section 3 gives a brief review of cloud computing; Section 4 addresses the CSCL platform in detail; Section 5 proposes an implementation for the platform and section 6 concludes this paper.

2. Related Work

The primary aim of CSCL is to provide an environment, mainly virtual environment, which supports collaboration between learners with the help of computer technology. Reviewing the evolving journey of computer mediation, it turns out to be many supporting platforms.

Centralized servers, including clusters, are regular platforms for hosting virtual learning environments, and they are always based on client-server model, browser-server model or web service model. These models can host a majority of web-based learning software, such as learning management system (LMS) [9-10] and learning content management system (LCMS) [11-12]. It is centralized and easy to manage on one hand, but inflexible and vulnerable on the other.

Peer-to-peer network based CSCL could make good use of distributed resources and leverage the advantages of ubiquitous computing [13-14]. Yet the efficiency is limited, and effective administration remains to be a challenging problem.

There are lots of grid-based learning platforms, e.g. GridCole [15], COLEG [16], GESOTC [17], ULabGrid [18], AKT (CoAKTinG [19]), the Knowledge-Grid-Based Cooperative Learning Environment [20], the Agent-Based Collaborative Virtual Environment for technology enhanced learning in the Service Grid [21]. Grid computing used to be an excellent resource management paradigm, especially with the help of OGSA and Globus Toolkit. It is outstanding in raw resource sharing, but cannot fully keep up with the dynamic changes of learner’s requirement these years.

Some researchers focused on learning through clouds. B. Dong et al [22-23] introduced cloud computing as the infrastructure of e-learning ecosystem and designed a sustainable framework. By integrating couples of social software or existing tools (e.g. YouTube, iGoogle, Matlab), several feasible solutions have been proposed on how to build a virtual and personal learning environment [24-27]. N. K. Richard [28] gave a general and detailed discussion toward a cloudy academy, which would indeed give insight to future work and greatly benefit higher education. North Carolina State University [29] developed a virtual computing laboratory which enabled learners to reserve and access VM with a basic image or specific applications environments (e.g. Matlab and Autodesk).

However, little attention has been particularly paid to manage CSCL onto clouds except F. Doelitzscher et al [30]. They introduced their research on building a private cloud named CloudIA, and also handled the
requirements of e-learning and collaboration in a university. Based on SVN and a XMPP based instant messaging server (Jabber), it could leverage the user to setup their own working groups and adding various functionalities. However, Encapsulation of working group was implemented by web-based software (Apache, PHP), which take place at IP level (Layer 3). By using the VLAN MAC frame, we can make this done at a lower level (i.e. Data Link Layer), which would promote the performance between learners.

3. Cloud Computing

As a new promising paradigm, cloud computing can offer utility-oriented IT services to users based on a pay-as-you-go model. In Berkeley’s View [4], “cloud computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased”. It can also make good use of economies of scale, and dynamically deliver/configure almost any IT related services on demand [5]. Moreover, it can conserve more energy, which is the ideal computing platform for the coming low-carbon economy [6-8]. With cloud computing platform, a number of useful features would be enabled, such as:

- **Massive scalability**: Based on virtualization technology, cloud computing can highly consolidate servers, and elastically provision resources on demand as needed. It can keep up with learner’s dynamic requirement, cut down running cost and conserve more energy.

- **High sharing & insulation**: By virtualization, resources within cloud platforms are highly shared between virtual machines (VMs) and learner on one hand; on the other, they are securely isolated from each other. Customized interfaces are supported to define the access level.

- **Efficient communication**: On cloud platforms, communications between learners are getting more efficient by taking advantage of virtual network. Within a specific virtual network, learners are closely connected with VLAN, which is more efficient than connection via Layer 3.

- **“Thin” client**: As a remote learning center, cloud platforms allow a variety of clients to access, including desktop, laptop, cell phone, embedded device and dedicated “thin” client. This enables learners to access and learn anytime and anywhere they want.

4. A Computer-supported Collaborative Learning Platform Based on Clouds

In order to maximize learner’s flexibility, every learner is entitled to have super authority over a given VM. CSCL is generally implemented through virtual networking technology. For efficiency purpose, we limit the size of a collaborative group to [3, 6]. Teacher’s role can be played by the combination comprised of the platform and the leader of the group, which can give multiple groups general instructions in parallel on one hand, and produce some special ideas for specific group on the other.

4.1. Architecture

The architecture of the platform is generally based on browser/server model, consisting four layers (See Fig.1). Communication between layers is done via web service interfaces and shared information (e.g. user profile, schedule policy) is housed by Database.

1. Fabric Layer. This layer lies at the bottom of the platform, serving as the basis of the whole architecture. Physically, it consists of several heterogeneous/homogeneous physical servers; however,
virtualization breaks down the physical barriers inherent in isolated resources, and automates the management of these resources as a single entity through hypervisor technologies [31]. Thanks to virtualization, this layer can provision virtual computing, virtual storage and virtual network services to upper layers.

2. Supporting Layer. Logically, this is where the “classroom” and learning contents located and merged. Making advantages of virtualization technology, this layer consists of two modules.

- Learning Pool: There are couples of VMs located here, which can be provisioned to learner through the management of LMS. Collaborative groups are logically presented in the form of VM groups, within which communications can be isolated and encapsulated. Learners are routed here to engage into a collaborative group.

- Data Center: Data center is of key significance to both the learners and the cloud provider. Customized VM images, learning contents are housed here.

3. Management Layer. This is a decision making layer, and it is in charge of the whole platform’s management, which also consists of two modules.

- LMS: This module fetches learning request from the web layer and schedule it based on some predefined policies. By making good use of learners’ profile and current situation of supporting layer, it either builds a new collaborative group or schedules one learning request to a specific collaborative group, and then keeps in touch with it in learning pool. LMS also accept on-demand dynamic reconfigurable request, and schedule it to appropriate VM. Dispatcher takes this result into mind and carries it out, including preparing VM images and learning contents.

- LCMS: LCMS receives requests from web portal and LMS module. It handles the request via a scheduler and a dispatcher. Scheduler can make some decisions (e.g. which project to provision?) based on learner’s profile (e.g. educational profile, age) and evaluation history in database. The dispatcher carries out the decision and provisions the content to the target group.

4. Web Layer. This is a web portal for learners to access services, who could engage in a collaborative group to learn later. This can be done via a desktop, a laptop, a cell phone or a specially desired thin client, only standard I/O devices and network connection is necessary. Through learner’s personal browser, learners can select the learning content, request for a new learning group or join an existing one.
4.2. Mechanisms

Based on the architecture, this platform can provide several basic mechanisms of CSCL (See Fig.2). For example, we suppose that one group (group members $M_a$, $M_b$ and $M_c$) is learning new words (new words $W_x$, $W_y$ and $W_z$), and $M_a$ servers as the leader.

1. **Self-learning**

   First things first, learners themselves have to learn a subset of (for efficiency) or the full set (for interconnectivity) of the requested learning content. And we assume everyone within the group have to learn a subset in the following. Splitting of learning content and provision to individual learner is done by the platform. Assume that $M_a$ gets $W_x$, $M_b$ gets $W_y$ and $M_c$ gets $W_z$ respectively, and this can save each member about 2/3 time.

2. **Discussion**

   When self-learning is done, the leader of the group could start the discussion process. Through discussion, different subsets of the learning content could be exchanged (i.e. $M_a$ gets $W_y$ and $W_z$, $M_b$ gets $W_x$ and $W_z$, $M_c$ gets $W_x$ and $W_y$); therefore, everyone gets a global view of the learning content without learning the full set. Discussion between group members can be carried out via a web service (such as bulletin board service) hosted by the leader, which can be accessed through the browser. Since the underlying implementation is based on VLAN technology, it is more efficient than regular browser-server models.

3. **Project**

   After self-learning and discussion, the group would be prompted to accomplish a project to consolidate learning. The project is based on the learning content. The fulfillment of the project cannot be reached without communication and teamwork, which means much more than just learning. The accomplishment of project is mainly directed by the leader through the discussion board, and carried out by group members on their own VMs.

   Take the new words learning for example, the project could be “make a list of differences between $W_x$, $W_y$ and $W_z$, and write a short story with what have been learned”.

4. **Evaluation**

   After receiving the final result (i.e. the project), the platform would send a feedback to the group. Evaluation is of high importance to both the learners and the platform. The evaluation is stored into the database by the platform. It can be an important reference for post settings, such as the splitting granularity of content, project level and the group size.

5. **On-demand responds**
Dynamic changes of learner’s requirement are fully supported by this platform. For instance, the leader of a learning group could add/remove a VM when the group is working. And this can be done on demand by the platform, and the final result would be released in seconds.

5. Implementation

We carry out a series of experiments to implement the platform discussed in previous sections. The testbed is composed of 4 personal computers (HP Compaq dc 7900), each of which has 4 cores (Intel(R) Core(TM) 2 Quad CPU Q8400 2.66GHz 2.67GHz), 4GB memory, and 300GB Hard disk, connected with 100Mbps switched Ethernet. The operating system is Ubuntu Server 9.10 AMD64. We select one computer to be the cloud scheduler (which housed LMS, LCMS and Database), and make the remainder as working nodes (Learning Pool and Data Center).

Through KVM, Eucalyptus (v2.0.1) [32] and Elasticfox (v1.7), we setup the base environment. Note that, Secure group within virtual network, which is based on VLAN, was firstly developed for private use, and multiple-user mode is not directly supported. Through key sharing, we can make the secure group work for our purpose. Although not secure enough, it is adequate for CSCL purpose.

For convenience, we have bundled two kinds of images, leader’s images and regular images. Within the leader’s image, two interfaces (i.e. communication between the platform and group members respectively) are implemented. Communication with the platform is done via a telnet service, whereas administration and
discussion with group members are through a bulletin board service (e.g. PunBB).

In the experiment, there are three learners (Jason, Jordon and Jackson) within a new word learning group. Subsets of learning content are \{hello\}, \{hi\} and \{halloo\}. Provision of content is implemented via a telnet service (See Fig. 3). When self-learning is finished, learners would log on to the discussion board, which is hosted by the leader, for discussion process. When everyone is online, the leader can start and host the discussion as well as the project (See Fig.4).

We implement these processes using two images respectively, e.g. Ubuntu server 9.10 AMD64 and Windows XP; each of which has two versions, e.g. leader’s image and regular image. For briefness, we just give two figures to illustrate the learning process, e.g. Fig.3 and Fig.4. The implementation show that, the average response time (from requested to ready to access) of learner’s VM has been limited to 15 seconds averagely. For on-demand mechanism, we make the leader of collaborative group add a VM instance when the group is working, and it takes about 12s on average to ready for new learners to log on.

6. Conclusion

As an indispensable ingredient of CSCL, computer mediation evolves through couples of phases, e.g. centralized server mode, peer-to-peer mode, grid computing mode. However, with daily rising trend on requirement’s dynamic changes in service, existing models fall short to respond on demand. In this paper, by taking advantage of cloud computing, we propose a feasible CSCL platform and give an implementation using Eucalyptus. The implementation shows that it can not only fulfill the basic requirement of CSCL, but also respond learner’s dynamic need on demand.

However, this platform cannot be directly applied in real practice yet due to the fact that the service level agreement (SLA) (e.g. business protocol [33]) between the learner and the platform has not been successfully set up. Next we would explore the particular scenario related to CSCL and exploit a suitable business model.

Acknowledgement

This paper was supported by the Shanghai Leading Academic Discipline Project (No. J50103), the Ph.D. Programs Fund of Ministry of Education of China (No. 200802800007), the Key Laboratory of Computer System and Architecture (Institute of Computing Technology, Chinese Academy of Sciences), and the Innovation fund of Shanghai University (No. B16010809007).

References


