International Conference on Complex, Intelligent and Software Intensive Systems

A Replication-based Approach for the Improvement of the Online Learning Experience in Distributed Environments

Santi Caballé[1], Fatos Xhafa[2], Ajith Abraham[3]

[1]Open University of Catalonia, Department of Computer Science, Multimedia, and Telecommunication
Barcelona, Spain
scaballe@uoc.edu
[2]Polytechnic University of Catalonia, Department of Languages and Informatic Systems
Barcelona, Spain
fatos@lsi.upc.es
[3]Center of Excellence for Quantifiable Quality of Service, Norwegian University of Science and Technology
Trondheim, Norway
ajith.abraham@ieee.org

Abstract

Modern on-line collaborative learning environments need to be continuously adapted, adjusted, and personalized to each specific target learning group. Moreover, these environments are to enable and scale the involvement of an increasing large number of single/group participants who can geographically be widely distributed, and who need transparently share a huge variety of both software and hardware distributed resources. As a result, collaborative learning applications need to be designed in a way that overcome important non-functional requirements arisen in distributed contexts, such as scalability, availability, interoperability, and integration of different, heterogeneous, and legacy collaborative learning systems. In this paper, an innovative distributed-based approach is presented for increasing the overall performance of collaborative learning systems that contributes to the effectiveness of the collaborative activities, such as online discussions. The experimental results show an outstanding effect on the learning processes and outcomes by enhancing and improving the learning experience a great deal.

1. Introduction

Over the last years, e-Learning, and in particular Computer-Supported Collaborative Learning (CSCL) [1], [2] applications have been evolving accordingly with more and more demanding pedagogical and technological requirements. In particular, collaborative learning environments must provide advanced enablement for distribution of both collaborative activities and the necessary functionalities and learning resources to all participants, regardless the location of both participants and resources. The aim is to enable the collaborative learning experience in open, dynamic, large-scale and heterogeneous environments [3], [4], [7]. From this view, one of the main challenges in the development of CSCL systems is to overcome important non-functional requirements arisen in distributed environments such as scalability, flexibility, availability, interoperability, and integration of different, heterogeneous, and legacy collaborative learning systems.

Little investigation [7], [8] has been conducted so far on how to take advantage of distributed infrastructure to directly impact on the learning processes and outcomes by means of meeting the above-mentioned non-functional requirements. Most of research on distributed learning deals with how to leverage computing, communication, and multimedia technologies to create richer, more flexible, and cost-effective distributed learning environments (DLE) than those standard lecture hall settings and give students access to a wide range of resources independently of place and time [9]. However, the issues of how to deal with current DLE performance problems are, to the best of our knowledge, not sufficiently investigated, such as, the enormous amount of requests, simultaneous connections of users, and server's response time [7].

From the experience at the Open University of Catalonia (UOC) certain requirements are especially frustrating when they are not fulfilled appropriately during the collaborative learning activity, such as fault-tolerance [4], scalability [5], and interoperability [8], which may considerably degrade the overall system performance. As a consequence, they may have considerable repercussions on the learning performance and outcomes as their lack impedes the normal learning flow as well as discriminates learners in terms of technology skills and technical equipment [10]. Indeed, some studies indicate higher student

433

IEEE computer society

dropout and course withdrawal rates in on-line settings [9] as well as poor student performance in terms of quantity and quality of participation [4].

In order to deal with these issues, distributed technology, such as Grid [6], has come to play a major role by capturing the vision of a networked computing system that provides broad access not only to massive information resources, but to massive computational resources as well. It is increasingly used for complex areas, which are computationally intensive and manage large data sets. The concept of distributed computing extends to a large-scale, flexible, secure, coordinated resource sharing among dynamic collections of individuals, institutions, and resources [3], [6]. These features form an ideal context for supporting and meeting the mentioned demanding requirements of collaborative learning applications.

In overall, system's fault tolerance is one of the most frustrating aspects during the on-line collaborative learning experience as it makes participants' requests be temporarily neglected in the presence of system failures [4]. In order to provide high levels of robustness and service availability, a learning system should seamlessly recover from failures and keep on providing service.

In this paper, it is shown how a distributed approach based on replication can provide permanent service and considerably improve the effectiveness of the collaborative learning experience. To validate the approach, several experiences were run using ad hoc collaborative learning tools to support the real online collaborative learning activities, such as in-class online discussion assignments occurring in the virtual learning context of the UOC. In these experiences both a centralized and distributed approach were used in terms of infrastructure and useful conclusions were drawn from comparing results achieved of both experiences.

The rest of the paper is organized as follows. Section 2 presents a centralized approach of an experience involving a real online in-class discussion and the results achieved while Section 3 shows the distributed approach in support for a similar experience. This section represents the core of this paper and presents the development decisions made to build an ad hoc discussion tool that took great advantage of distributed infrastructure. Finally, Section 4 presents the results and their analysis of this experience of using distributed infrastructure in support for a collaborative learning activity, such as an online discussion. The paper concludes by summarizing the main ideas of this contribution and outlining ongoing and future work.

2. A Centralized Online Discussion Approach

An experience of using a discussion tool called Communities of Learning Practice Environment (CoLPE) (see [10] for a complete overview of this tool) took place at the UOC involving 86 graduated students enrolled in the course Methodology and Management of Computer Science Projects. Students were equally distributed into two classrooms and participated in the experience at the same time. Students from one classroom were required to use the standard asynchronous threaded discussion forum offered by the virtual campus of the UOC while the other group of students used the new CoLPE outside the campus to support the same discussions with the same rules during the same time.

The whole experience consisted in carrying out a class assignment in the form an on-line discussion activity for 3 weeks. The students enrolled in the course were free to open discussion threads at convenience where they proposed strategies, ideas, etc., to appropriately deal with the topic of to the discussion (i.e, "Change management: necessity or virtue?"). Any student could contribute in a discussion thread as many times as needed so as to provide new argumentations with regards to the issue addressed. The only requirement was to submit at least one post to any thread.

The whole experience was supported by a Zope server [12] on the server side, which run on a single node (i.e., Linux SuSE 2.4.21-99 machine, Intel Pentium 4 CPU 2.00 GHz, 256MB RAM). A statistical analysis of the results of the discussion comparing both the standard and CoLPE tools is shown in Table 1.

Table1. Main statistics results from the class assignment using both discussion tools.

Statistics	Standard tool	CoLPE
Number of students	43	43
Number of threads	29	17
Total of posts	174	93
Mean number (posts/thread)	M=6.0 SD=2,7	M=5,5 SD=4,5
Mean number (posts/student)	M=4,0 SD=1,6	M=2,2 SD=3,8

In particular, the problems were originated as follows. First, Zope is a powerful server that demands a fairly amount of hardware resources to run. Second, during the rush hours, the growing number of users

who concurrently requested CoLPE's knowledge-related data-intensive functionalities (see Figure 1) generated noticeable performance repercussions and system crashes on the central server. Finally, the server was not available once for a few hours during the rush time due to maintenance of the internal network.
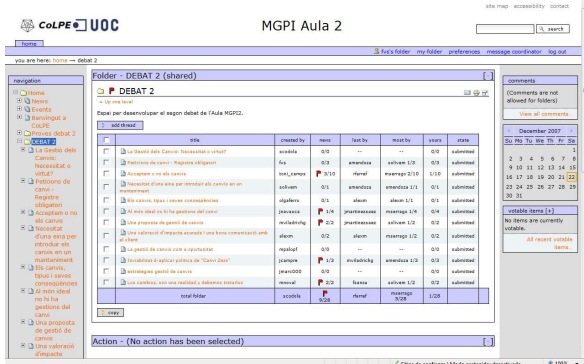

Figure 1. Partial feedback at folder level presented to all participants. The *most by* and *yours* indicators allow students to compare their own quantitative performance to the rest of the contributors. Flags indicate new contributions.

As a consequence of this centralized approach, important non-functional requirements could not be completely satisfied in terms of availability, scalability and performance. Indeed, a negative impact on the discussion process was caused by the lack of fault-tolerance and performance repercussions caused by the demanding knowledge-based functionalities, which involved large amounts of complex information about group activity to be processed (see Figure 1). To this end a distributed approach is proposed next to improve the overall performance of the discussion process.

3. A Distributed Approach in Support for the Online Discussion Process

In this section a similar experience to that of the previous section is reported. In this case, another ad hoc discussion tool called Discussion Forum (DF) was used (see [11] for a complete review of this tool) developed by the same authors' research group to support an in-class discussion assignment at the UOC. In this experience, a replica-based distributed infrastructure was added during the whole discussion process and the results achieved were analyzed in the same terms as those of the experience reported on in the previous section.

3.1. A Discussion Tool for Distributed Environments

In developing the DF, special attention was paid to distribution, flexibility, robustness and availability as key aspects to address the current needs for meeting the more and more changing and demanding non-functional requirements in the CSCL domain in general and specifically in online collaborative discussions.

In order to meet these requirements, the development of the DF was based on a generic, robust, interoperable, reusable, component-based and service-oriented platform called Collaborative Learning Purpose Library (CLPL) (see [13] for a complete description of this platform), This platform implements the conceptualization of the fundamental needs existing in any collaborative learning experience. In addition, the CLPL is highly flexible in distributed environments in terms of implementation languages and underlying software and hardware platforms, which represent a suitable context to guarantee robustness, scalability and service availability [14].

Following the principles and software architecture of the CLPL, the primary principle to fulfill the online discussion functionalities designed in the DF was to provide a broad set of independent fine-grained Web-services grouped by a particular purpose, such as the authentication process and the presentation of the knowledge extracted from the discussion activity. The goal was both to enhance the flexibility in the development of the DF and to ease the deployment of this application in a distributed environment.

To this end, each particular behavior of the DF is discomposed into three specialized Web-services matching each of the three layers of a typical software development, namely user interface, business and data [15]. As a result, the completeness of each specific behavior goes through three separate, necessary, sequential steps that connect to the client on one side and to the persistent storage (e.g., database) on the other side.

For instance, the authentication process is formed by three different, independent Web-services, namely the authentication user interface, the authentication business, and the authentication data. Thus, when the user tries to log in the DF, the client code calls the authentication user interface web-service, which is in charge of collecting the credentials presented by the user. Then, this web-service calls the authentication business web-service so as to verify the correctness of the user's input (e.g., input no blank, well-formatted, etc.). Moreover, as part of the business process, this web-service validates the users' input upon the information existing in the database by calling the

authentication data web-service, which is responsible for acceding the database and extracting the authentication data of the user.

A clear, independent, and separated vision of each single behavior of the DF into fine-grained task-specific Web-services results in a natural distribution of the application into different nodes in a network. This distribution is driven by matching the web-service's purposes and both the node configuration and location in the network. According to this view, the web services in the user interface layer should be located nearby the client; the business Web-services would be better suited if allocated in those nodes with high-performance processors, and, finally, the data Web-services could be attached or located nearby the database, which is to be supported by nodes with high storage capability. Moreover, in this context, the database can be also distributed as it is clearly separated from the data Web-services, which would be in charge of updating and keeping the consistency of the different instances of the database.

The work methodology proposed by the CLPL offers the DF total flexibility as to where (i.e., network node) to install both each DF function (i.e., DF behavior) and each layer for this function (i.e., web-service). Moreover, the widely adopted standards of the Web-services technology (e.g., HTTP and TCP/IP) help communicate the Web-services with each other in a network just using their IP address and passing through firewalls and other barriers that other technologies have problems to overcome. In addition, there exist many open-source technologies that deal with Web-services, such as Apache Tomcat and Axis, allowing the DF to easily use and deploy their services.

In the context of distributed infrastructures, both the independence between the fine-grained services provided by the DF and the use of replication produce many important benefits. Indeed, by installing and deploying replicas of the Web-services all over the network, high levels of service availability are easily achieved by redirecting a request to an exact replica of the web-service when a node is down. Concurrency and scalability become natural in this context by parallelizing the users' requests and using as many replicas as necessary. Finally, interoperability between replicas is inherent in this context as Web-services are fully independent from hardware platforms and programming languages.

To sum up, combining the replication approach, the Web-services technology, and leveraging distributed infrastructure, the realization of the most demanding non-functional requirements existing in online discussions becomes a reality.

3.2. Adding Distributed Infrastructure

The Discussion Forum prototype is currently supported by three nodes located in two separated buildings of the UOC (see Figure 2):
- Linux Red Hat 3.4.6-3 cluster, Intel Xeon CPU 3.00 GHz 4GB RAM
- Windows 2003 server, Intel Pentium 3 CPU 800 MHz 512MB RAM
- Linux SuSE 2.4.21-99 machine, Intel Pentium 4 CPU 2.00 GHz, 256MB RAM
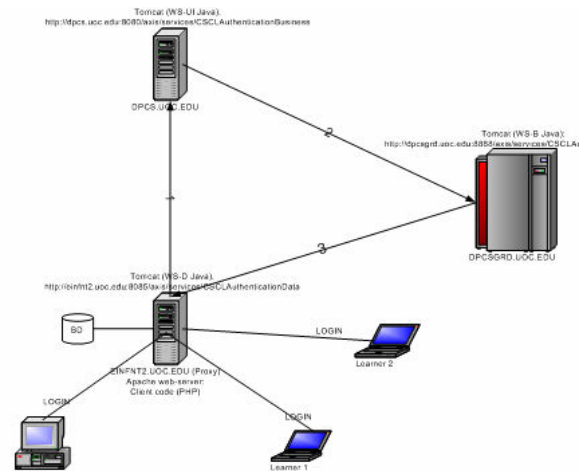

Figure 2. Distribution of the DF functionalities into 3 nodes. Each node is in charge of a fine-grained Web-service representing a specific functional layer (i.e., user interface (UI), business (B), data (D)).

For the purpose of this experience, all the Web-services of the DF prototype were replicated on each node. Moreover, the same client code in the form of PHP running on Apache Web servers was installed in two nodes (Windows server and Linux SuSE machine). Finally, in this prototype, just a single instance of the database was installed in Windows server. The Windows server acted also as an entry proxy by redirecting at HTTP level all the requests received to either itself or the Linux Red Hat cluster. In this first version of the approach the database is supported by just one node, which makes the system fully dependent from it. In future iterations of this approach, it is planned to distribute the database in several nodes and manage its consistency by the data Web-services. The ultimate goal in this initial version was to prove the feasibility of the distributed approach.

To that end, (see Figure 2) upon the reception of a user's request, the Windows server proxy first pings at Linux SuSE machine whether it is alive. If so, the Linux SuSE machine starts dealing with the request by executing its PHP code, otherwise the Windows server

itself is doing so by executing its own PHP code. From the client PHP code it actually starts the sequential call chain of Web-services for each layer, namely the user inter-face, business, and data Web-services for each function requested. Each call to a web-service implies, if possible, to send the current request to another node. However, before calling a web-service on a different node a ping is sent to check the node's availability. If the other two possible nodes are down, the node managing the current web-service calls the next web-service locally and tries again to find another node where to call the web-service of the next layer. When the request gets finally the data layer (i.e., the data web-service), the call is addressed from any node to the Windows server. Once the information has been successfully managed in the database, the response is sent back to the client through the same way the request took.

4. Experimental Results

In order to validate the addition of distributed infrastructure to the DF and analyze its effects in the collaborative learning activities in general and specifically in the online discussion process, an experience was run involving 80 graduated students distributed into 2 classroom in the course Methodology and Management of Computer Science Projects at the UOC. The experience consisted in carrying out a discussion process in the form a discussion for 3 weeks involving all the students. Students were equally distributed into two classrooms and participated in the experience at the same time. Students from one classroom were required to use the well-known asynchronous threaded discussion forum offered by the virtual campus of the UOC while the other group of students used the new DF outside the campus to support the same discussion with the same rules during the same time. The discussion in both groups consisted of discussing the same issue: project management requirements vs. product requirements. The discussion procedure was very similar to that of the experience reported previously (see Section 2).

From the pedagogical point of view, the experience resulted very successful as it showed the benefits of providing an adequate information and knowledge management in supporting the discussion process. Indeed, the quantity and quality of the contributions during the discussion greatly increased in comparison to the experiences achieved using the well-known asynchronous threaded discussion forum offered by the virtual campus of the UOC (see Table 2).

Statistics	Standard tool	DF
Number of students	40	40
Number of threads	57	65
Total of posts	171	549
Mean number (posts/thread)	M=3.0 SD=2,4	M=8,4 SD=5,0
Mean number (posts/student)	M=4,2 SD=1,9	M=13,7 SD=3,1

Table2. Main statistics extracted from the discussion using both discussion tools.

Despite both the number of participants and type of discussion as well as the functionality available were quite similar to the previous experience (see Section 2), the results improved according to both the participants' and tutor' point of view. Indeed, the system performed smoothly and just one time the DF was reported to be unavailable. This improvement came mainly from the utilization of other nodes apart from the Windows server, which was still overused. This fact provided an important performance gain that all students appreciated a lot and influenced on the discussion process in terms of more contributions of better quality in average (see Table 3).

Indicators	Centralized approach	Distributed approach
Tutor assessment 0-10 (average)	6.2	7.8
Peer assessment 0-10 (average)	5.4	6.5
Participation impact (average)	+1.8	+4.1
Passivity (pending to read on average)	88.3%	31.9%

Table 3. Main learning indicators extracted from both experiences

Table 3 shows a comparative study between the centralized (CoLPE) and distributed (DF) experiences. Certain key indicators, such as the tutor assessment and the participation impact, improved a great deal, which showed the effects of the distribution approach in the learning process. Particularly interesting is the improvement of the passivity indicator showing the contributions in average pending to read. The reason may be found on the normalization of the system's performance, which allowed the participants to spend time reading others' contributions. This, in turn, enhanced the discussion process by increasing the cogniscitive level of the participants in the discussion.

5. Conclusions and Future Work

In this paper, an innovative replicated-based approach has been presented for increasing the overall performance in terms of service availability of collaborative learning distributed systems. This approach contributes to the improvement of in-class collaborative learning assignments, such as online discussion activities. The results of the experiences reported in this paper show an outstanding effect on the learning process by enhancing and improving the learning experience a great deal. In particular, the improvement in terms of robustness, concurrency, scalability and other non-functional aspects of the discussion system actually impacted positively on the discussion process itself.

These experimental results should, however, be taken carefully as more validation process needs to be undertaken. Nevertheless, the results show the suitability of adding distributed infrastructure to overcome important issues, such as the lack of service availability, arisen during the online discussion process, which impede the normal learning flow.

There is still a great deal of work to undertake so as to evolve appropriately the distributed version of the discussion tool prototype. However, this initial approach encourages us to work towards this direction. In the near future, it is planned to deal with the complex issue of distributing the database into the available nodes of the distributed infrastructure so as to avoid any central point of failure. Moreover, this initial approach is to be extrapolated by deploying the discussion tool in the nodes of PlanetLab platform so as to validate it in a real and complex distributed environment.

6. REFERENCES

[1] Koschmann, T., Paradigm shifts and instructional technology. In T. Koschmann (Ed.), CSCL: Theory and Practice of an Emerging Paradigm, Mahwah, New Jersey, Lawrence Erlbaum Associates, (1-23), 1996.

[2] Dillenbourg, P, Introduction; What do you mean by "Collaborative Learning"? P. Dillenbourg (Ed.), Collaborative learning. Cognitive and computational approaches, 1-19. Ox-ford: Elsevier Science, 1999.

[3] Pankatrius, V., Vossen, G., Towards E-Learning Grids: Using Grid Computing in Electronic Learning. Proceedings of IEEE Workshop on Knowledge Grid and Grid Intelligence, Halifax, New Scotia, Canada, (pp. 4-15), 2003.

[4] Caballé, S., Xhafa, F., Daradoumis, Th. A Service-oriented Platform for the Enhancement and Effectiveness of the Collaborative Learning Process in Distributed Environments. Proceedings of the GADA 2007. Vilamoura, Algarve, Portugal. LNCS 4804. pp. 1280-1287. ISBN: 978-3-540-76835-7, 2007.

[5] Yang S., Jiang, J., Shi, M. A Scalable Framework for Large-Scale Distributed Collaboration. In proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design. IEEE Computer Society. Pp. 1-6 ISBN: 1-4244-0165-8/06/, 2006.

[6] Foster, I. and Kesselman, C. The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann, San Francisco, CA, pp. 15-52, 1998.

[7] Bahrami, K., Abedi, M., Daemi, B. Implementation of Distributed E-Learning System on Power Line Network. In Proceedings of the Third Advanced International Conference on Telecommunications (AICT'07), pp. 29-35. IEEE Computer Society. ISBN: 0-7695-2843-0/07, 2008.

[8] Bahrami, K., Abedi, M., Daemi, B. A Web Service-Based Portal Framework for Distance Learning on Power Line Network. In Proceedings of the 6th International Conference on Information, Communications & Signal Processing, 2007, pp. 1-5. IEEE Computer Society. ISBN: 1-4244-0983-7/07/, 2007.

[9] Alavi, M. Distributed Learning Environments. Computer. 37(1) pp. 121 – 122. IEEE Computer Society, 2004.

[10] Caballé, S., Feldman, J., Thaw, D. Supporting Communities of Learning Practice by the Effective Embedding of Information and Knowledge into Group Activity. In proceedings of the International Workshop on Collaborative E-Learning Systems and Applications (CESA 2008). IEEE Computer Society. ISBN: 0-7695-3109-1/08, 2008.

[11] Caballé, S., Daradoumis, Th., Xhafa, F. Providing an Effective Structured and Context-aware Asynchronous Discussion Forum for Collaborative Knowledge Building. In proceedings of the World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2008). AACE Press. ISBN: 1-880094-62-2, 2008.

[12] Zope: http://www.zope.org (October 2008).

[13] Caballé, S., Daradoumis, Th., Xhafa, F. A Generic Platform for the Systematic Construction of Knowledge-based Collaborative Learning Applications. Book: Architecture Solutions for e-Learning Systems. Chp. XII, pp. 219-242. Hershey, PA, USA: IGI Global. ISBN: 978-1599046334, 2008.

[14] Shum, S.B., CoAKTinG: Collaborative Advanced Knowledge Technologies in the Grid, Proc. Second Workshop on Advanced Collaborative Environments, Eleventh IEEE Int. Symposium on High Performance Distributed Computing (HPDC-11), Edinburgh, Scotland, 2002.

[15] Object Management Group: Model-Driven Architecture http://www.omg.com/mda (October 2008).