Peer-to-Peer Communication for Computer-supported Collaborative Learning

The PeCoCC Framework

Mais Hasan
Communication Networks Group
Technische Universität Ilmenau
Ilmenau, Germany
Mais.Hasan@tu-ilmenau.de

Jochen Seitz
Communication Networks Group
Technische Universität Ilmenau
Ilmenau, Germany
Jochen.Seitz@tu-ilmenau.de

***Abstract*— Computer-supported collaborative learning is an important domain of e-learning dealing with researching efficient methods to encourage the people to learn together with the help of their computers. The learning environments used in this domain are usually client-server based solutions with some extra functionality needed for the collaboration between the members. Because of its architecture, which is similar to the collaborative learning network, the peer-to-peer (P2P) technology is suggested as a better solution. This paper introduces a P2P-based framework for applications in the area of computer-supported cooperative learning. This framework embodies a platform for P2P learning applications and consists of four different layers. It includes different P2P modules and chooses the module that works best according to the application requirements. The paper shows the advantage of the proposed framework.**

Keywords-Peer-to-peer communications; computer-supported collaborative learning.

I. Introduction

During the last decade, online learning has gained enormous interest in most educational institutes. E-learning can be defined as the process of using electronic media, information and communication technologies in education. E-learning includes numerous forms of educational technology in learning and teaching and can be used jointly with the conventional face-to-face learning. E-learning can occur in or out of the classroom. It is suited to distance and flexible learning and can be asynchronous or synchronous. As a result of the rapid improvement in the areas of education, information and communication technologies, various e-learning forms have evolved. This evolution started with using the information technology in the Computer Based Training (CBT) and develops in the direction of exploiting the internet and social interaction in the Virtual Learning Environments (VLE) and Computer-Supported Collaborative Learning (CSCL) applications [1].

The most used learning environments were based on the client/server approach. A server is the source of services and information, several clients have access to. However, the approach suffers from two main problems: scalability and single point of failure. Thus, different approaches to overcome these problems have been developed. One of these is a paradigm shift to the P2P model. In this approach, the communication partners act as server and client at the same time. They all offer a part of the information and retrieve information from other nodes known as peers. The more peers take part in a P2P communication, the better this network scales and the higher its reliability is. Several application fields have utilized this P2P approach so far. In this paper, we introduce a framework to apply this approach for computer-supported collaborative learning.

Therefore, the paper is organized as follows: Section 2 shortly deals with computer-supported collaborative learning and reviews some CSCL-tools based on P2P technology. Different P2P technologies and their properties are analyzed in Section 3. Section 4 presents our designed CSCL-tool, the **pe**er-to-peer **co**mmunications for **c**omputer-supported **c**ollaborative learning (PeCoCC) framework and its functioning. Section 5 gives an overview of the current state of the work and summarizes the paper.

II. PEER-TO-PEER COMPUTER-SUPPORTED COLLABORATIVE LEARNING

Computer-supported collaborative learning is an emerging branch of e-learning allowing several students to cooperate with each other and with the teaching staff online in order to solve shared tasks or to exchange their skills. Computer-supported collaborative learning is related to collaborative learning and Computer Supported Cooperative Work (CSCW). By collaborative learning we generally mean that a group of students work together to discuss, solve or evaluate teaching materials; on the other hand, computer supported cooperative work addresses the technologies and tools supporting people in their work. Hence, computer supported collaborative learning refers to the use of CSCW- technologies and tools by a group of collaborative students in a learning process. These technologies and tools have been developed to provide an efficient learning process. Woodill [2] gives an overview of all the different technologies used to support collaborative learning (see Figure 1).

As mentioned above, one of the suggested approaches to overcome the problem of the client-server model in collaborative learning is using the P2P technology. There is already some research work done in this area. Some educational P2P applications like, e.g., COMTELLA, EDU-

 ISBN: 978-1-61208-328-5

COSM, Edutella, and Groove have been developed for some specific needs and they are still under development

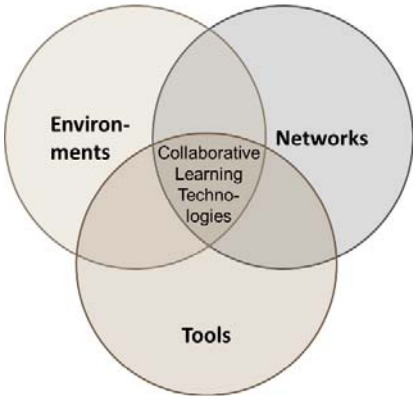

Figure 1. Information technologies used to support collaborative learning [2]

COMTELLA is a P2P file sharing system that allows students to contribute and share class-related resources with their community [3]. The shared papers are annotated with respect to their content in categories. COMTELLA uses a modified version of the Gnutella P2P protocol and instead of sharing the actual files, only their URLs are shared. A list of the shared articles, their URLs in the web and the corresponding comments are distributed among the users. There is one list for every category. If a student searches for a paper, he should only search the list of the matching category. Students can view and read the papers without downloading them by clicking on the "Visit" button in the COMTELLA user interface, which starts the default browser with the URL of the paper.

EDUCOSM is a web-based learning environment providing a shared view to the Web [4]. It consists of a collection of server-side scripts and an HTML and JavaScript based client that runs inside a web browser. The role of the server is to store the data and act as a proxy between the client and the rest of the web. The principles of EDUCOSM and COMTELLA are similar with the difference that the storage of the data in COMTELLA is distributed among the users.

Edutella is an educational P2P network built on Sun Microsystems JXTA Framework [5]. Edutella is an open source P2P application for searching semantic Web metadata. It uses the resource description framework (RDF) for presenting information in the web. Edutella deals with metadata about content, not with content itself. It adds a search service to the JXTA framework, so that any node that carries metadata about some resource can announce an Edutella search service to the network. The nodes in Edutella have actually at least one of three types of roles: provider (provides a query service), consumer (asks questions) and hub (manages query routing in the network).

The previously mentioned three P2P collaborative systems offer only one collaborative tool, mostly a file-sharing application, which is not sufficient for efficient collaboration among the users. These systems suffer the absence of a coordinative tool like a group calendar which is typical for team or group software. They also do not support cooperation applications like a whiteboard or a text editor.

These problems have been tried to be solved in one of the popular collaborative environments, Groove. It is a collaborative groupware based on the principle of a shared workspace [6]. Tools like a shared browser, a shared drawing board or a file archive are used to operate in this shared workspace. Groove provides servers that are used to detect new peers in the network and to store content if one or more peers are offline and cannot see the changes made at that time. Using server-based services threatens the availability of these services if one of these servers fails.

Groove is targeted at small workgroups and has its own protocols. It is only available for the windows platform, so it suffers interoperability problems. This manifests the need for a collaborative environment providing many collaborative and coordinative tools, supporting interoperability, and basing on fully distributed server-independent P2P communications and services. However, there is no open source software having the mentioned functionalities available at the moment.

III. PEER-TO-PEER TECHNOLOGIES

In contrary to the client-server model, all the members of a P2P network are equally offering and requesting services. Generally, we can assert that every P2P network is established on an overlay network, mostly based on Transmission Control Protocol (TCP) or on Hypertext Transfer Protocol (HTTP) connections. Thus, the overlay and the physical network can be separated completely from each other. Hence, the overlay connections do not reflect the physical connections. Nevertheless, it is possible to match the overlay to the physical network if necessary. P2P networks can be divided into two classes: unstructured and structured P2P networks.

In a structured P2P network, the network topology and the location of content is determined by employing a P2P protocol. In these networks, the content and the participating nodes share the same address space which makes it easy and expeditious to reach any content in this space. Structured P2P networks are based on a Distributed Hash Table (DHT) and have no central entities. Frequent signaling traffic is necessary to maintain the network awareness of the nodes. Pastry, Chord and Content Addressable Network (CAN) are examples for this class.

The distribution of nodes and content in unstructured P2P networks is executed randomly. The position of content can only be resolved in a local proximity of a node and only by flooding the request to a particular extent. In this way, these networks consume the bandwidth, which has been saved by their random distribution. Unstructured P2P networks can be centralized with an index server like Napster, hybrid with dynamic super nodes like Gnutella 0.6

and JXTA, or pure without any central entities like Gnutella 0.4 and Freenet [7].

IV. THE PECOCC FRAMEWORK

Until now, most P2P collaborative environments are developed for specific needs and a central entity is used in most of them. Therefore, we have developed a P2P framework for computer-supported collaborative learning, which we called PeCoCC. The PeCoCC framework uses different P2P overlays to support different applications. This characteristic of the PeCoCC framework enables completely separate working of the applications, which increases the robustness of the system. The PeCoCC framework has a layered architecture depicted in Figure 2.

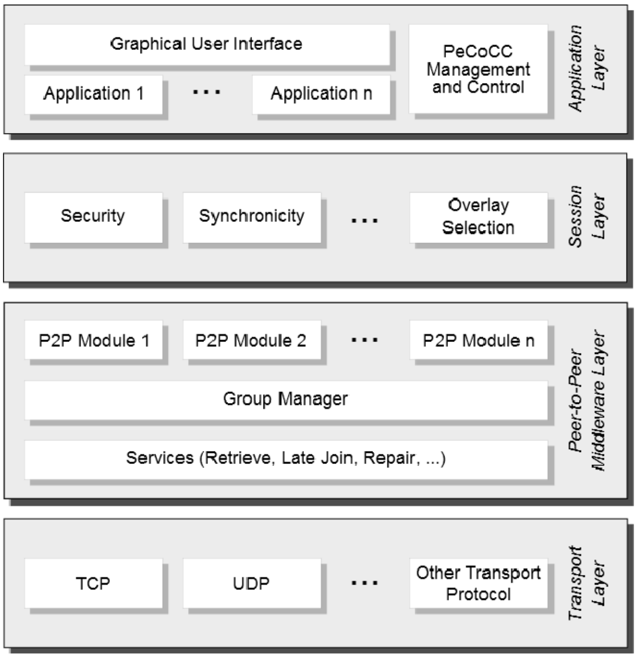

Figure 2. The PeCoCC framework

The main functions that this framework has to provide are as follows:

Collaborative Tools – the PeCoCC framework provides three applications, which are important to cooperate efficiently. A shared calendar will be used to allow the users to organize their regular meetings; a distributed text editor can be used to jointly make notes on a given subject or to brainstorm about a topic and P2P file sharing allows users to access the distributed contents they need to cooperate.

Group Management – The framework must include functions to manage the communication among collaborative group members.

Recovery – The framework implements a "late join" mechanism to consistently provide the information for latecomers to enable them to participate in the ongoing session. This is typically achieved by getting the state of the distributed application from the current participants and by initializing the application of the latecomer with this information.

Synchronization – For some application modules (e. g., distributed text editor), the group members need to be synchronized to interpret the events in the correct time and order.

Security – The framework provides security mechanisms (e. g., encryption) to keep personal data integer and secure.

The PeCoCC framework consists of four different layers, which support its main functionalities. The layers are introduced in the next subsections.

A. Application Layer

The application layer consists of three main parts. The graphical user interface allows the interaction between the user and the framework. It facilitates a consistent operation of all the desired CSCL services. The PeCoCC Management and Control (MC) module is responsible for controlling the data flow through the framework and the work flow among the users. It influences the application program and also saves a list of the cooperating participants, their application-dependent roles and their priorities. The peers should be identified by the MC module to be allowed to enter the system. The rights of the users can be defined by the applications themselves. In a file sharing application, for example, all the users have equal rights, while in a text editor, the teacher should have more possibilities to manipulate the entered information than the students. Furthermore, in sessions without a dedicated chair, the MC module is responsible for defining the user that has the according rights of a chair.

The third part is composed of application modules. These can be freely chosen and added on demand. As stated above, the first modules of our choice are a distributed text editor application, a calendar, and a file sharing application. These modules interact with the graphical user interface and the MC module.

B. Session Layer

The session layer provides general mechanisms that are necessary for the offered applications. Currently, we have concentrated on two mechanisms. The security module is responsible for securing private user data and the synchronicity module, which provides a mechanism to synchronize the different group members so that all of these receive the events in the same order. This service is necessary for real time applications like a distributed text editor. Since the using of P2P overlay in the PeCoCC framework is application dependent, the session layer includes the Overlay Selection (OS) module, which is responsible for saving the information about the appropriate P2P overlay for each application.

C. Peer-to-Peer Middleware Layer

As P2P technologies exist with respective advantages and disadvantages, the PeCoCC framework allows the usage of different pure P2P technologies. Each technology is

 ISBN: 978-1-61208-328-5

encapsulated in a P2P module and offers its communication functionality. Which module should be used is selected by the OS module in the session layer according to the needs of the application. To illustrate the functionality of this layer, we have started with two well-known P2P approaches. The content addressable network (CAN) can be used for an efficient distribution of information and teaching materials. It is a structured P2P network based on DHT. CAN offers high scalability and reliability and provides more load balancing than any other pure P2P overlay [8], but it does not take into account the underlying network conditions. Therefore, it is not suitable for real-time applications due to the fact that it does not make any correlation between the overlay distance and the actual number of unicast hops between the hosts in the underlying network. PASTRY, another structured P2P network considers the underlying network topology and supports a scalable and distributed object location and routing in application layer [7]. The PASTRY protocol can thus be integrated in a P2P module for applications like a distributed text editor and instant messaging.

Furthermore, in each network the users belonging to one user group have to be managed. This is done in the module called group manager. This module is responsible for forming and supervising a collaborative user group. Finally, this layer comprises a set of services that extend the P2P modules with late join, retrieval and repair functions.

D. *Transport Layer*

The transport layer provides access to different commonly used transport protocols. These are selected accordingly to the requirements of the applications. For example, the distributed text editor utilizes the Transmission Control Protocol, which provides a reliable transport service.

E. *Functioning of the System*

When the user starts one of the available applications (e.g., a distributed text editor), the MC module is activated and sends a CHOOSE message to the OS module in the session layer containing information about the opened application.

The OS module then decides on the basis of the opened application which P2P module is more appropriate for this application and replies to the MC module in the application layer with an OVERLAY message. The OS module also activates the necessary services for the opened application.

The MC module receives the OVERLAY message and retrieves the saved list of the expected participants (the participants of an application should be previously registered by the MC module and saved in a specific list). The MC module then sends a START message to the corresponding P2P module in P2P middleware layer.

In the P2P middleware layer, the selected module starts the P2P connection and sends a DISCOVERY message to find out if the P2P network has already been built or not. If it gets an answer, it will take part in the P2P network. The group manager then sends a JOIN message with the group ID using the overlay multicasting protocol to join a collaborative group and retrieve the information about the participating peers as well as the important data to interact in the current session. If a collaborative group does not exist, the group manager will send a CREATE message in the P2P network to create a collaborative group with a specific group ID.

The information retrieved by the group manager from the members of the group is returned to the MC module to specify the role and the rights of the peer in the session. The role of the peer will appear in form of active or inactive interaction possibilities in the user interface.

Which rights and roles the user has, is managed in tables and saved in the MC module. One table is defined for each application. In the case where all users have the same rights, a mechanism to manage and identify the roles will be used. This mechanism can take into account the registering sequence of the participants or the alphabetic order of the user ID names, etc.

V. CONCLUSION AND FUTURE WORK

In this paper, an overview about the current P2P collaborative environments and their use case has been presented. The P2P technologies have been briefly highlighted. We have also introduced our PeCoCC framework to allow computer-supported collaborative learning based on pure P2P networks that provide fully distributed and server-independent P2P communications and services, which increase the availability of these services and solve the problem of single point of failure present in server-based systems. The PeCoCC framework is still in implementation phase. It supports the interoperability and is being implemented in Java using the integrated development environment eclipse. A distributed text editor is being built on the top of the PASTRY algorithm and is using the PASTRY overlay multicasting protocol SCRIBE. To evaluate the performance of the framework, a P2P simulator named Peerfactsim.KOM will be used. This simulator has a similar layered architecture like the PeCoCC framework and supports many forms of messages to communicate among the layers in the host. It is also implemented in java and it offers a user-friendly graphical user interface [9]. In the future, a file sharing application will be built on the top of the content addressable network P2P algorithm. A mechanism for structured metadata will be needed to fulfill the educational purposes in P2P networks.

REFERENCES

[1] J. Moore, C. Dickson-Deane, and K. Galyen, “e-Learning, online learning, and distance learning environments: Are they the same?,” Journal of Internet and Higher Education, vol. 14, Mar. 2011, pp. 129-135, dio:10.1016/j.iheduc.2010.10.001

[2] G. Woodill, “Computer Supported Collaborative Learning in Education and Training: Tools and Technologies,” Phil. Trans. Brandon Hall Research. San Jose. CA. USA, 2008.

[3] J. Vassileva, “Harnessing P2P power in the Classroom,” Proc. 7th International Conference, ITS, Aug. 2004, pp. 305-314, doi:10.1007/978-3-540-30139-4_29

[4] M. Miettinen and J. Kurhila, “EDUCOSM – Personalized Writable Web for Learning Communities,” Proc. Information Technology: Coding and Computing, ITCC, Apr. 2003, pp. 37-42, doi:10.1109/ITCC.2003.1197496

[5] C. Qu and W. Nejdl, “Interacting the Edutella/JXTA Peer-to-Peer Network with Web Services,” Proc. IEEE Symp. Application and the Internet, 2004, pp. 67-73, doi:10.1109/SAIT.2004.1266100

[6] http://technet.microsoft.com/enus/magazine/2006.10.intothegroove.aspx

[7] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, “A survey and comparison of peer-to-peer overlay network schemes,” Communications Surveys & Tutorials, IEEE, vol. 7, 2005, pp. 72-93, doi:10.1109/COMST.2005.1610546

[8] D. Boukhelef and H. Kitagawa, “Efficient Load Balancing Techniques for Self-organizing Content Addressable Networks,” Journal of Networks, vol. 5, Mar. 2010, pp. 321-334, doi:10.4304/jnw.5.3.321-33

[9] https://sites.google.com/site/peerfactsimkom/?p=413