

VIDEOCONFERENCING WEB APPLICATION FOR CARDIOLOGY DOMAIN USING FLEX/J2EE TECHNOLOGIES

Imen Debbabi and Ridha Bouallegue

Innov'COM Research laboratory, University of Carthage, Tunis, Tunisia.

debbabi.imen@gmail.com
ridha.bouallegue@gnet.tn

ABSTRACT

The recent advances in computer technology and data networking have made videoconferencing system a popular medium for users to interact with one another from remote locations. This system offers communication between more than two users, who are able to interact through their webcams, microphone and other components. The use of this system has been increased recently due to many reasons, for one thing, progress in Internet access in different networks like companies, universities and houses, with the increase of available bandwidth whereas the decrease of delay in sending and receiving packets. On the other hand, the coming of Rich Internet Applications (RIA) means that a large part of web application started to be implemented on the web browsers. This paper discusses the conception of multiparty videoconferencing systems using technologies of Web 2.0. For our conceptual Videoconferencing Platform, we have developed many feature : live audio video, text chat, video recording, user and room management and quality control. Videoconferencing modules have been carried out using open source technologies Flex and J2EE.

KEYWORDS

Videoconferencing System, RIA, web2.0, Flex, Red5, J2EE .

1. INTRODUCTION

Telemedicine is defined as the use of telecommunication and information technologies in order to provide clinical health care at distance. It helps eliminate distance barriers and can improve access to medical services that would often not be consistently available in distant rural communities using two-way videos, emails, smart phones, wireless tools and other forms of telecommunication technology [1].

It may be as simple as two health professionals discussing medical problems of a patient and seeking advice from over a simple telephone to more complex transmission of electronic medical records of clinical information, diagnostic tests such as E.C.G, radiological images etc . and carrying out real time interactive medical video conference with the help of Information Technologies.

This paper describes a videoconferencing system using open source technologies for a group of cardiologists connected through internet. There may be a situation where a cardiologist in a remote hospital may need to contact an expert to take his advice about a better treatment of a patient. Videoconferencing enables a group of cardiologists to organize meetings, conferences,

training or other distant meetings while giving the impression of being all present in the same room.

In the recent years, the videoconferencing systems have become more complex due to increasing demands of this type of systems in the internet and the heterogeneity of terminals, such as laptops, mobile phones, tablets, etc with a variety of network access connections including ADSL, cable Modem, Wi-Fi, UMTS, etc. On the other hand, the maturity of multimedia codecs, formats and protocols has become a reality in the last decade. Therefore, a lot of videoconferencing applications and solutions arise every year.

I.debbabi and al. [2] give a survey of the videoconferencing standards and videoconferencing solutions and compare them. These systems are provided in two different ways. The first type is the **centralized system**. Generally, drawbacks of the centralized system are not scalable and the operations of the MCUs (Multipoint Conference Unit) are very complex and expensive. The outgoing bandwidth of the server is shared by all the concurrent participants. Specifically, the more clients there are, the lesser the bandwidth each participant can have. The more expensive multipoint control units are capable of handling more connections, the faster data transfer rate is and the more participants can be displayed on the video screen at one time. In addition, MCU requires a higher bandwidth to disseminate a single video signal among participants. The second kind of videoconferencing system is the **multicast videoconferencing system**. It is often free of charge and easy to install and use, although quality cannot be guaranteed.

In the traditional systems, videos, audios and document information encoding, decoding and transfer are implemented by the developers, which makes the systems more difficult to be developed, and the deployment and maintenance more expensive. These Desktop applications have a difficult installation, maintenance and access flexibility for many reasons like the increasing number of clients with different hardware and operating systems. Web developers have always tried to build one type of client which is richer, more powerful and sensitive and have more interesting visual characteristics than the traditional HTML. The appearance of RIA technology makes it possible to deploy rich client program as easily as to use the web on the Internet.

In this paper, we present a technical overview of technologies for Rich Internet applications, Red5 Media Server, J2EE framework and Flex technologies. We also present architecture of videoconferencing system using these technologies. The remainder of this paper is organized as follows: Section 2 presents some technologies of RIA. Section 3 presents the design of our system. In Section 4, we present the realization of the system. Section 5 describes the conclusion and future work.

2. TECHNICAL OVERVIEW

2.1 Technologies for Rich Internet application

The term "Rich Internet Applications" was introduced in a Macromedia whitepaper [3] to designate a unification of desktop and traditional Web applications whose goal is to take the advantages of both architectures and overcome their disadvantages. "Macromedia" define RIAs as combining the best user interface functionality of desktop software applications with the broad reach and low-cost deployment of the web applications and the best of interactive multimedia communication.

Typically, a RIA is a web application running in a web browser as part of a web site and is loaded by the client along with some initial data, then it manages the data rendering and event

processing, communicating with the server only when the user requires further information or must submit the data. However, this type of application shares several characteristics like a rich user experience and complex user interfaces with traditional desktop applications, and can be accessed using only the web browser with the use of JavaScript; via a browser plug-ins or sandboxes. It forces the users to install a plug-in before launching the application. Once installed, the application runs inside a sandbox delivered within the plug-in.

Before viewing a list of Rich Internet Application frameworks, we should know that a framework is a collection of software libraries providing an application programming interface (API), which provides generic functionality that separates them from other normal libraries.

In the case of RIAs, there is an extensive list of software frameworks that can be useful in different contexts, with different programming languages and operating systems. The vast majority of frameworks are intended to run over almost all the operating systems.

These Frameworks are Adobe Flash, Microsoft Silverlight, Oracle Java FX, AJAX or HTML5. They have to be installed on clients that are going to run these applications.

Many studies regarding the installation made depict the market penetration and global usage comparing different technologies. They show that Adobe Flash, Microsoft Silverlight and Java are installed in more than 60% of machines.

2.1.1 Adobe Flash Player

Adobe Flash (formerly Macromedia Flash) is a multimedia platform used to add animation, videos, and interactivity to web pages. Flash is frequently used for advertisements and games. More recently; it has been positioned as a tool for the Rich Internet Application .It is mainly based on an object-oriented language named *ActionScript*.

In order to run the Flash content, the users need to install the Flash Player plug-in previously. Nowadays, Flash content may be displayed on various computer systems and devices using Adobe Flash Player, which is free for common web browsers, some mobile phones and a few other electronic devices (using Flash Lite).When Flash Player starts running the application, it manipulates vector graphics to provide animation to various components, like text, drawings, images, and UI objects. It also provides advanced APIs, such as the possibility of manipulating images directly, audios and videos. It supports multimedia streaming applications, such as videos and audios streaming as well as bidirectional multimedia communication and can capture the user's input via a mouse, a keyboard, a microphone, and a camera.

2.1.2 Real Time Messaging Protocol (RTMP)

The protocol called **Real Time Messaging Protocol** (RTMP) was initially implemented by Macromedia in 2002 and now it is owned by Adobe. It was first used to stream the content across the internet from the servers to the clients which were applications running in a web browser plug-in (Macromedia Flash Player). It was then used to implement web conferences, by accessing the user's camera and microphone through the same clients, and then sending them using RTMP to send and receive media flows. In 2009, having acquired Macromedia, Adobe decided to publish an open specification of this protocol in [4].

According to those specifications, the RTMP protocol has multiple variations, i.e. the "plain" RTMP protocol; RTMPS which is RTMP over a TLS/SSL connection, RTMPE which is RTMP

encrypted using Adobe's own security mechanism, and **RTMPT** which is encapsulated within **HTTP requests** to go through firewalls [4].

RTMP works on top of TCP which maintains persistent connections and allows *low-latency* communication. It originally follows client-server architecture by defining requests and responses. This protocol splits media streams into fragments and their size is negotiated dynamically between the client and the server. Thus, it transmits as much information as possible with a little overhead.

RTMP defines several virtual channels to provide multiplexing of audios, videos and data sent in the same TCP connection. It defines headers like a timestamp, size of packet, id of the channel, as shown in Fig 1.

At higher level, RTMP encapsulates MP3, AAC and Speex audio, FLVI and H.264 video, and can make remote procedure calls (RPCS) using a specific format for the representation of the data, named Action Message Format (AMF)[4].

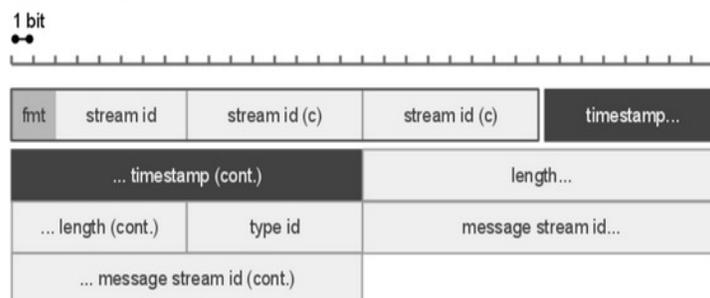


Figure 1 : RTMP packet structure

2.2 Red5 Media Server

The development of Flash-based applications requires the implementation of a centralized Server architecture based on a multimedia server, which redirects all the traffic sent between clients, for example **Adobe Flash Media Server**. This server is a new platform used for communication between users. It integrates Flash multimedia interactive features, and adds real-time audio, real-time video and real-time data streams and other new features like broadcasting. This server is proprietary media server from Adobe Systems but not free. In this work, we need an open system server. There is an open source technology, called **Red5 Media Server**, which delivers a powerful video streaming and multi-user solution to the Adobe Flash Player and other exciting client technologies [4].

Red5 is an open source software package written in the Java language. Its purpose is to consolidate communication between Adobe flash applications. It offers the same functionality as **Adobe Flash Server**. It supports live stream publishing, audios and videos streaming, object sharing as well as the recording of streams [6, 7].

By using this platform, you can save recorded audios, videos from a network, share data objects, and transfer the audios, videos and shared data objects with multiple clients. Therefore, it stands as a solid solution for businesses of all sizes, such as enterprises and it is widely used for developing videoconferencing systems, multi-user gaming and enterprise application software. Red5 includes a support for the latest multi-user APIs such as **NetConnection**, **NetStream** and **SharedObject** while providing a powerful RTMP / Servlet implementation. In addition to

supporting the RTMP protocol, the application server has an embedded Tomcat Servlet container for JEE Web Applications.

In Red5, we developed a browser based videoconferencing application. Here, the client uses Adobe Flash Player to capture videos and audios from players continuous media clips from a web server. This platform integrates communication and application functions, via a flash player on the client, captures and shares audios, videos and data streams. By using Red5 and Flex, we can easily create a real-time communication system, which allows two or more end-users to communicate instantly. For example, we can use the Red5 to easily create video conferencing systems, video telephone system and video chatting systems [6, 7].The system presented here is designed based on Flex. It uses extended UI component library and defines rich user interface on MXML.

Macromedia Flex is a representing server and application program framework, which meets the needs of enterprise-class programmers who hope to develop RIA and run on J2EE. It can run in the application server and provide standard-based, declarative programming methods and processes, run-time services, application integration and management capabilities.

Flex can integrate Java object access or xml using the existing code and information. Furthermore, Flex can also be integrated with some of the existing presentational technology and framework, such as JSP, Struts, etc [8].

In addition, it deals with procedure logic by using script towards objects, and runs **swf** client program translated by Flex server into Flash Player. Combining with audio, video and some real-time communication technologies, the RIA has highly interactive user experience.

2.3 J2EE Framework

J2EE is a set of specifications, which define the standard for developing multi-tier enterprise applications with Java. This platform provides a complete framework for design, development, assembly, and deployment of Java applications built on multi-tiered distributed application model. This specification defines numerous API services and multiple application programming models for developing applications and integrating them with the enterprise systems as shown in Figure 2.

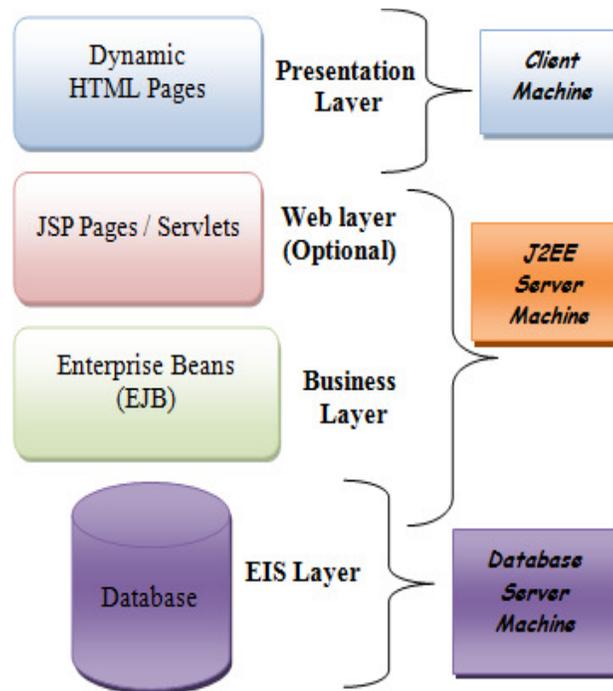


Figure 2: Architecture of J2EE framework

A J2EE application is divided into multiple layers, as seen in Figure 2. Applications can have three or four layers, although most of them only have three [8, 9].

The top layer exists on the client's machine called The **Presentation Layer**. It consists mainly of interactive webpage interfaces, but occasionally contains some functionality. This layer is responsible for the management of the user's request and for making the corresponding response. Example of the frameworks is **Struts** [9].

The next level is the Web tier. This level may contain Java Server Pages (**JSPs**) and **servlets**, which dynamically handle requests to the server. The Web tier is usually excluded, in which case the Client tier interacts directly with the **Business layer**. This latter layer holds the enterprise beans, which retrieve and process the data from the database.

This layer is responsible for dealing with application procedure business logic and business checking, and responsible for the management services. An example of this framework is **Spring**[9].

Finally, the Enterprise Information System (EIS) layer is the database in which the information is stored. This layer is responsible for building a relationship between the java domain objects and the database tables. An example of this framework is **Hibernate** [9].

The Business layer is responsible for dealing with application procedure's business logic and business checking, and responsible for the management services. An example of this framework is **Spring**.

2.4 Integration of Flex and J2EE

Prior to the release of Flex, J2EE developers could tightly integrate their middleware with a Flash-based client by using the Flash Remoting technology. This was introduced by one of the authors in the **Macromedia Press title Reality J2EE—Architecting for Flash MX**.

As shown in Fig.3, Flex is integrated in the Presentation Layer. Flex communicates with the **Business Layer** through the AMF gateway. The advantage of this model is that it can quickly and conveniently integrate the Flex into the existing J2EE framework without affecting the original structure [8].

Introducing Flex into data services will affect the performance of Flex. Flex offers several ways to send data: calling the Java classes loaded in Flex classpath; sending a request to the web proxies service or HTTP server [8]. Flex data services can be divided into Web Service Agent, remote object proxy and HTTP services, etc. Two better choices are the XML of HTTP Service and AMF's Remote Object [8].

In this framework, a statement which appeared in the Flex page announces that a Remote object (FlexFacade type) run on the server-side. Flex is responsible for the data transfer between Flex pages and Java objects. We need to define a corresponding method in an object when the Java method is necessary in all the Flex pages which will call the corresponding method in Flex Facade object, and then the method will call the Java method which realizes the business logic in the background [8].

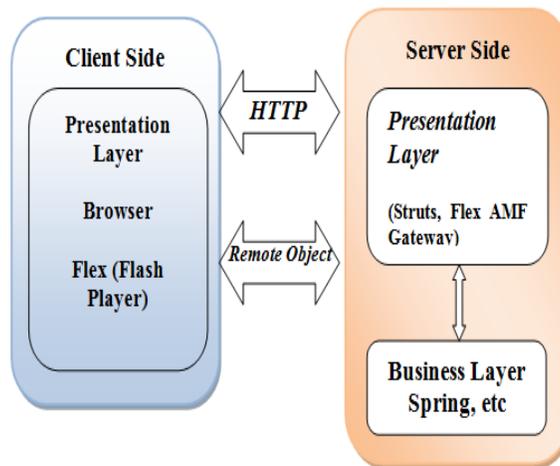


Figure 3: Integrable framework of Flex/Struts

3. SYSTEM DESIGN

3.1 Design of Videoconferencing system

In recent years, more and more multimedia conference systems have been deployed on the Internet. A typical Videoconferencing System has the following functionalities : Live Audio / Video, Text Chat and Video Recording .Firstly, the live audio / video tool supports multiple video displays, showing the video streams from the cameras focusing on the different remote cardiologists. Next, the Text Chat facilitates chatting amongst cardiologists sitting in remote place. Finally, the video Recording helps automatically record and save all the videos in a central archive, where cardiologists can access them conveniently when they need to review them.

Our videoconferencing prototype has four main modules as demonstrated in Fig. 4.

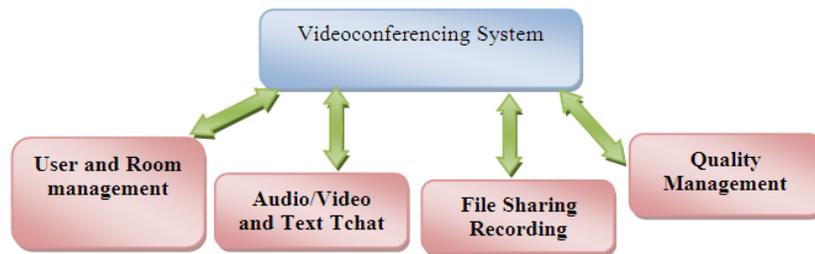


Figure 4: Prototype Videoconferencing System

- **Audio/Video and Text Tchat Module:** This module captures the videos, relays it to the receiver's end to enable cardiologist sitting in remote place chat with one another.
- **File Sharing and Recording Module:** This module helps the users share files and record their video/audio conferencing and text chatting through the internet.
- **User and Room Management Module:** The purpose of this module is the authentication and identification of the registered users. It also facilitates the connection between the user groups for audio/video conferencing.
- **Quality Control Module:** The role of this module is to control the quality of images and audios and capture the videos and relays them to the receiver's end with a desired quality of service.

3.2 Implementation of Videoconferencing system

The architecture of our application is client-server, where a computer interacts with others on the Internet.

As we mentioned earlier, our application consists of three parts: User and Room management developed in Jsp / Servlet / JavaBeans, some for streaming videos developed in Flex (MXML, Action Script) and a portion for controlling a PC remotely with suitable tools like VNC viewer open sources and VNC server.

To do so, we need:

- **On the application server:** *Apache Tomcat* is used as an application server for the management of the rooms and users.
- **On the streaming server (Red5):** It is responsible for disseminating all the information Multimedia (video, micro, slides) to all the participants.
- **On the MySQL database server:** handles the data storage.
- **A thin client:** a web browser (Google Chrome, Firefox ...)

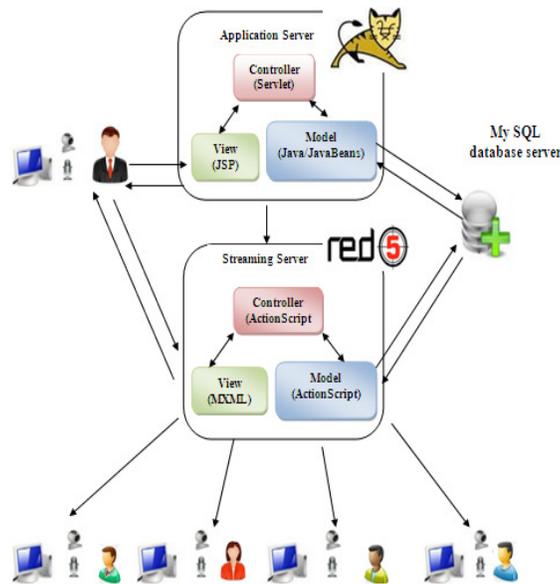


Figure 5: Architecture of application

In the realization of our project, we opted for MVC architecture to provide insurance maintainability, scalability of the application and the speed of development. The Figure 5 shows the architecture of our application.

4. REALIZATION

The implementation of the proposed architecture in this paper has resulted in the provision of a system that consists of a set of classes and methods that we will detail in this section.

4.1 Several important classes and methods

4.1.1 Class Application

Application class contains information about a **Red5 Server** application. It maintains this information until the application program is unloaded.

There are several important events in the application class:

- **AppStart:** it is called when the applications are loaded.
- **AppStop:** it is called when the applications are uninstalled.
- **Connect:** it is called when a client connects with the application.
- **Disconnect:** it is called when a client disconnects with the application.

There are several important methods in the class Application:

- **acceptConnection:** it accepts the connection from a client to an application procedure.
- **broadcastMsg:** it announces the news to all the connected clients.
- **disconnect:** it disconnects the connection between the server and the clients.
- **rejectConnection:** it refuses an application to another.

4.1.2 Class *NetConnection*

NetConnection is used for the management of a two-way connection between Flash Player and red5 Server. It can let you connect to the remote Flash object and red5 Server, while red5 Server allows you to use macromedia Real-Time Messaging Protocol to share audio, video and data information.

4.1.3 Class *NetStream*

NetStream realizes a data stream connection between Flash Player and Red5 Server. A NetStream object is just like a channel of a NetConnection objects. This channel can, by using the publication method of NetStream object, publish audio and video data, or by using Play() method of NetStream objects, subscribe a published flow or data. When necessary, we can also, using NetStream object, publish or broadcast live video and audio information, or play previously recorded video and audio information. There are two important methods in NetStream object: NetStream.attachAudio() and NetStream.attachVideo(). These two methods are essential for the network video call. We can make use of these two methods to bind the audio from the microphone, and the video from the camera with a network stream.

4.1.4 Server-side programming

Service-side scripting can be used to control log-in processes, and control the events in connection Micromedia Flash program which determine the data that the users get from the Flash program. Server-side script file is generally main.asc. When the system is running, the file is placed in the Applications directory of the Flash Media Server. Service-side scripting main code is as following:

```

userList=[];
application.onConnect = function(currentClient)
{
application.acceptConnection(currentClient);
currentClient.communicateServer= function(value)
{
currentClient.username=value;
userList.push(value);
trace("The current user list"+userList);
application.broadcastMsg("playOtherVideo",userList);
}
}

```

4.1.5 Client Programming

The release and acceptance of video and audio signals is realized by NetStream object of red5 Server. It mainly constructs a TCP-based connection by using the RTMP agreement, between the client and FMS. A connection can contain several RTMP data circulation channels. When the clients log in video conferencing systems successfully, FMS uses a data flow channel to publish the user's video and audio streams. At the same time, FMS, by using several data flow channels, receives other client user's video and audio stream.

The following is the registered user' video, audio stream core code, published by FMS. First of all, the client to establish connection:

```

var RTMP = " rtmp ://localhost:5080/Videoconference" ;
_global.out_nc = new NetConnection () ;
out_nc.connect (RTMP ,PhoneID) ;
var mic = Microphone.get() ;
mic.setRate (11) ;
mic.setUseEchoSuppression (t rue) ;
mic.setSilenceLevel (0 , 0) ;
Publish the audio stream of our side to the user:
outAudio_ns = new NetStream(out_nc) ;
outAudio_ns.attachAudio(mic) ;

```

```

outAudio_ns.publish("Audio" + PhoneID) ;
Publish the video stream of our side to the user:
outVideo_ns =new NetStream(out_nc) ;// New data streams
outVideo_ns.attachVideo (Camera.get()) ;
outVideo_ns.publish("Video" + OutID) ;
Accepting the video stream of other side:
inVideo_ns = new NetStream(out_nc) ;// new data streams.
Video_AV.attachVideo(inVideo_ns);//
inVideo_ns.play("Video " + PhotoNum_mc.CallOutNum) ;

```

4.2 Comparative study and discussions

Traditional video conference systems have problems of development, expensive deployment and maintenance. Flex and Red5 Server provide a new and better solution to develop videos, webcast, MP3 streaming audios, videos conferencing and audio and video chat. This application combines the best user 's interface functionality of desktop software applications with the broad reach and low-cost deployment of Web applications and the best of interactive, multimedia communication.

Among the different applications of videoconferencing proposed in literature we have described only a few examples in this section.

Ponec et al. [11] considered P2P multi-rate multi-party conferencing systems. In a multi-rate setting, different receivers in the same group can receive video at different rates using layered videos. In particular, their study focuses on issues related to multi-rate setting in comparison to the case of single rate videos. It studies the optimal usage of peer uplink capacities of P2P utility maximization in a multi-rate multicast which provides a novel multi-tree formulation.

Nefsis [10] provides dedicated cloud computing resources for video conferencing. The users automatically connect to geographically close servers distributed on the Internet to have a low-latency experience.

We now consider well known, free or commercial conferencing systems such as Skype, Gmail chat and Apple's iChat, and discuss how they address the challenge of Multipoint conferencing. Skype provides multipoint audio conferencing, but video conferencing in Skype is only point-to-point. Recently, the developers have added a group feature which is limited to 5 persons; however, it is not known what the bandwidth requirements are.

GMail chat has a video conferencing module (Google gmail chat), but it works only for two persons. Moreover, it uses a special hardware provided by Vidyo.

Apple's iChat provides multipoint video conferencing; however, one of the peers has to have enough download and upload bandwidth to initiate the conference where it presumably acts as a software MCU (Multipoint Multipoint Control Unit : Apple iChat).

Many other popular online chatting applications (like Skype, Msn, Yahoo, messenger, Google talk, etc.) support multi-party videoconference but they have not a good quality of service therefore, they are not utilizable for professional use like in telemedicine and therefore not considered here.

We have considered Tree applications, of which we list the maximum bandwidth, the delay, the technique of video coding, the maximum number of simultaneous conference participants, and the architecture (S/C or P2P) they belong to Table 1.

From Table 1, we observe that all the applications support only a very limited number of participants and the applications that support more than 10 simultaneous participants, who all use a centralized or P2P network structure.

Table 1 : A comparison of videoconferencing solutions

	Delay in ms	Bandwidth kbps	Number of Video participants	Video coding	Architecture
Ponec and al.,[11]	<100	2000	15	H.264/AVC	Distributed P2P
Nefsis [10]	20	14Mbps	8	-	centralized
Our application	<100	2500	8	-	centralized

We observe that the maximum bandwidth is 14 Mbps which corresponds to cloud computing solutions whereas the second bandwidth is **2.5 Mbps** which corresponds to our web application. The solutions based on P2P architecture features of the best bandwidth and the best delay is inferior to **100 ms**.

3. CONCLUSIONS

Videoconferencing systems have problems of development and expensive deployment. In this paper, we provide a new method using open source technologies based on the Flex and J2EE video conferencing network system to develop a low cost one. It is intended to support consultations for a group of cardiologists connected through the internet.

For future work, Videoconferencing System with 3G and 4G mobile telephony and virtual reality technologies can be developed.

REFERENCES

- [1] <http://www.americantelemed.org/about-telemedicine/what-is-telemedicine>.
- [2] I.Debbabi,"A Survey of multimedia videoconferencing system and a proposal for a novel hybrid cloud and P2P architecture". ISSN 22773061. 1641 | Page. Aug 20, 2013.
- [3] J. Duhl. White paper: Rich Internet Applications. Technical report, IDC, November 2003.
- [4] Adobe Systems Incorporated, Adobe's Real Time Messaging Protocol specification , 21 December, 2012.
- [5] <http://www.red5.org/>

- [6] Liu Lu and Dong XiaoGuo, 2010. Red5 flash server analysis and video call service implementation. In Proceedings of IEEE symposium on web society, pages 397-400.
- [7] X. C. Peng, "The Integration of Flex RIA with J2EE Application," Application Technology and Research, June, 2008, pp. 37-39 (in Chinese).
- [9] <http://java.sun.com/j2ee>.
- [10] <http://www.nefsis.com> Nefsis. Online: <http://www.nefsis.com/How-Multipoint-Conferencing-Works/index-multipoint-video-conferencing.html>.
- [11] Ponc, M., S. Sengupta, M. Chen, L. Jin and P.A. Chou, 2009. "Multi-rate peer-to-peer video conferencing: A distributed approach using scalable coding. Proceedings of the IEEE International Conference on Multimedia and Expo, Ju. 28-Jul. 3, IEEE Xplore Press, New York, pp: 1406-1413. DOI: 10.1109/ICME.2009.5202767

AUTHORS

Imen debbabi was born in Monastir ,Tunisia, in 1984. She received the B.S. degree in Computer engineering from National Engineering School of Sousse, Tunisia, in 2008, and a master degree of "Intelligent and Communicating Systems "from National Engineering School of Sousse, Tunisia, in 2010. She is currently pursuing Ph.D degree in Higher School of communication, Tunis, Tunisia. Her research interests include, Multiparty Video Conference and multimedia System .

