

A Novel Distributed Architecture for Video Conference in LAN

Kai Shuang, Chunhua Wu, and Zhenjiang Dong

Abstract—This paper focuses on the video conference architecture in local area network (LAN). Based on the comparison of logic controlling, cost, hardware and network bandwidth among three different architectures and additionally taking the abundant bandwidth of LAN into account, we have testified that the server-forwarding is the proper architecture to implement video conference system in LAN for small-scale team.

Index Terms—Video conference system, LAN, distributed architecture, server-forwarding.

I. INTRODUCTIONS

Video conference system is a kind of communication system that mainly transmits visual information, which connects two or more multimedia terminals via network, and transmits video, audio and data signals between the terminals. It makes people who are far away together by hearing voice and seeing images, feeling like “talk face to face”. With the rapid development of computer network and multimedia technology, video conference system based on IP is becoming more and more attractive.

In this paper, we focus on the research of the best architecture to implement video conference system in LAN. The video conference system we desired should have the following characteristics: able to control the system without complex logic; implemented mainly by software, including the signal control and media stream processing; universal hardware device instead of the expensive and dedicated hardware facilities; low cost so that the small-scale team can afford. As we know, video conference system requires sufficient bandwidth, but usually there is enough bandwidth in LAN.

This paper studies three different architectures applied in video conference system. Giving the theory analysis of advantage and disadvantage of these architectures, then we select an optimal one to implement the video conference system in LAN. The first architecture is called Centralized. Under this kind of architecture, the system requires one or more multipoint control unit (MCU) to distribute media streams including audio, video and data [1]. The second architecture is called P2P. There is no centralized component

under this architecture, both control message and media data are directly transported between peers [2]. The last one is called server-forwarding architecture. There is a central server with this architecture which can manage users and sessions and more importantly, responsible to forward the media stream.

II. POSSIBLE ARCHITECTURES

These three architectures differ a lot. Different architecture has different component, different way of logic control and different size of media stream. In this part, we describe the difference briefly.

A. Centralized

Under this architecture, MCU receives the stream from all conference clients, extracts the information such as audio, video and signaling, then push the data into processing module, completing processes including audio, video mixing and switching, data broadcast and routing, timing and conference control. Finally MCU combines and sends all the information to corresponding terminal. The function of MCU requires complex control logic and powerful processing ability, so generally the price of MCU is often very expensive, which increase the cost of the system.

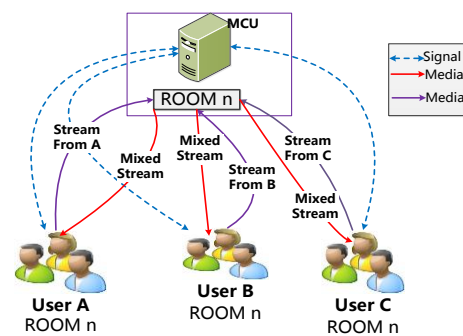


Fig. 1. The architecture of centralized.

As we can see from Fig. 1, clients exchange signal information with MCU, it is the MCU's duty to manage the conference room and all the users in the room. Every user sends its own stream to MCU and then MCU processes these media, mixing and switching finally sends it to corresponding user. No matter how many users in a conference room, a participant will only receive one media stream from MCU.

In this architecture, the MCU takes care of everything including media process and session control. That makes MCU in deep pressure especial for thorough media process. But this price makes client especially simple to implement and the network will also benefits from it. This architecture is efficient for big company who is willing to pay for this price.

Manuscript received November 3, 2014; revised June 20, 2015.

Kai Shuang is with the Beijing University of Post and Telecommunications. He is also with the Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory, China (e-mail: shuangk@bupt.edu.cn).

Chunhua Wu is with the Beijing University of Post and Telecommunications, China (e-mail: mail4wch@126.com).

Zhenjiang Dong is with the ZTE Cloud Computing & IT Research Institute, Nanjing 320100, China (e-mail: dong.zhenjiang@zte.com.cn).

B. P2P Architectures

There is no centralized component under this architecture, both control message and media data are directly transmitted between peers. However, it is rather complex to control and manage the video conference system under this architecture, and it requires higher performance ability on the clients' terminal.

As we can see from Fig. 2, there is no central unit under this architecture. Every client needs to not only connect with each other but also to maintain all the information of conference. Stream data from every client will be copied and send to others.

This architecture is like the opposite of the centralized one. Everything is in the control of the clients themselves. No one will do anything for them. If any client wants to join the conference, it needs to negotiate with others and take care of the media process by themselves. So the control maybe a little complicated and the process requirement to a client is relative high.

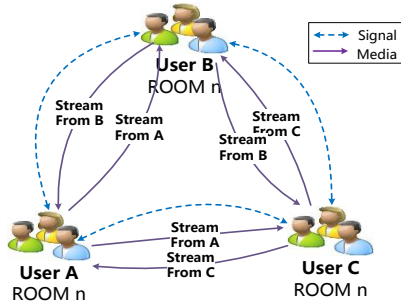


Fig. 2. The architecture of P2P.

C. Server-Forwarding

Under this architecture, it is the central server's duty to control the conference. Each conference participant sends its media stream to the server and then the server forwards the stream to other participants. Therefore in this process the server will require plenty of bandwidth, however there is not strict demand on processing ability of server (see Fig. 3).

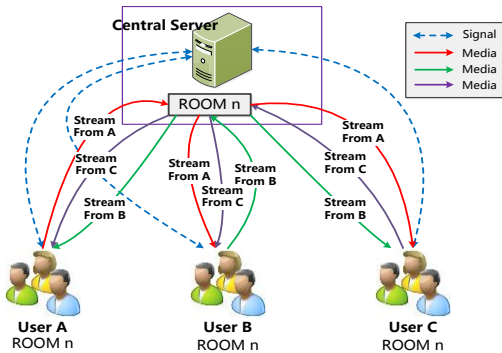


Fig. 3. The architecture of server-forwarding.

Same as MCU, central server needs to manage conference and users. Each conference participant sends its media stream to the central server and receives several copies of stream derived from other participants.

Server-forwarding architecture is a compromise to the centralized architecture and the P2P one. There exist one MCU-like server, which helps the session of the clients, but it will not go too far. The media still goes for the clients. In this

way, unlike P2P architecture, the control is quite simple and unlike centralized architecture, the central server will not be too heavy loaded. It is efficient and will be easy to implement.

III. THEORETICAL ANALYSES

These architectures differ on cost, control complexity, hardware and network bandwidth requirement. This chapter will analyze the respective advantages and disadvantages of these three architectures and combine the characteristics of LAN, then choose the most appropriate architecture to implement video conference system.

A. Network Bandwidth

Video conference system mainly transmits audio and video stream. The bandwidth required by uncompressed audio stream depends on sampling rate, sampling bits and channel numbers. The bandwidth required by uncompressed video stream depends on the pixels, color bits and frame rate. For the convenience of analysis, we assume the average size of media stream from terminals is MB/s, and there are N participants in the conference room.

In centralized architecture, every client sends the compressed media stream to MCU, MCU receives all the stream and processes them. Then MCU sends the mixed and switched media stream to corresponding terminal. The size of mixed stream differs with different MCU products, we assume it to be K , K must be far less than $N \times M$. The upload size of terminal is M , and the download size is K . And the size of media data received by MCU is $N \times M$, the data size sent by MCU is $N \times K$.

In P2P architecture, each client needs to send its own media stream to other $N-1$ conference participants, at the same time also need to receive media stream from other $N-1$ -conference participants. Therefore the size of media stream sent by each client is $(N-1) \times M$, the size of media stream received by each client is $(N-1) \times M$, the sum is the bandwidth required by each terminal.

In server-forwarding architecture, each conference participant sends his media stream to the central server, and then central server forwards the stream to other participants. For each terminal, the upload size of media stream is M , and the download size is $(N-1) \times M$. For central server, the size of media data received is $N \times M$, the size of data sent is $(N-1) \times N \times M$. totally, bandwidth required by central server is $N \times N \times M$. with the increase of the participants, the bandwidth need by central server increases at square level. We summarize the bandwidth cost as in Table I.

TABLE I: BANDWIDTH COST OF DIFFERENT ARCHITECTURE

Architecture	MCU Receive	MCU Send	Client Receive	Client Send
Centralized	$N \times M$	$N \times K$	K	M
P2P	0	0	$(N-1) \times M$	M
Server-forwarding	$N \times M$	$(N-1) \times M$	$(N-1) \times M$	M

B. Hardware

Here we mainly take requirements on the CPU and memory of computers into consideration.

In centralized architecture, MCU is the most important part of the system. MCU is mainly responsible for processing and

mixing audio and video, which requires high processing capacity. Under this architecture, the client just sends one copy of media stream to MCU and receives one copy of mixed media stream from MCU, so the size of media stream would not increase largely while new user enter into the conference room. Therefore general computer can meet the requirement of clients.

In P2P architecture, each conference participants need to send its own media data to all the other participants, and will accept media stream from all the other participants. Therefore, every client must maintain $2 \times (N-1)$ connection, and deal with $2 \times (N-1)$ copies of the media data. With the expansion of the conference, linear increase of terminal computing requirement may become the bottleneck of system.

In server-forwarding architecture, each conference participant sends his/her media stream to the central server. Then central server just needs to forward the stream to other participants without any processing. The central server has no too many computing tasks, so general computer can handle. As to client, each one needs to send its own media data to the central server, and will accept $N-1$ copies of media stream from MCU. So as same to P2P, the demand for computing ability is also linearly increasing with the increase of the number of participant. We summarize the hardware cost as in TABLE II.

TABLE II: HARDWARE NEED OF DIFFERENT ARCHITECTURE

Architecture	MCU	Client Hardware
Centralized	high processing capacity	low
P2P	No need	Relative high
Server-forwarding	Relative high	Relative high

C. Control Complexity and Cost

In both centralized and server-forwarding architecture, there is certain component to manage the conference room and participants, control logic is mainly concentrated in one unit (MCU or central server), at the same time the control logic on client is relatively simple. As to P2P, every participant needs to maintain all the information of conference room and the state of conference must be synchronized at any time. So the operation on client is rather complex. As for cost, the centralized architecture needs MCU, which is professional device and expensive. The server-forwarding architecture needs a central server to manage and forward media stream. But according to the analysis above, general computer can meet the requirement of processing ability of central server. There is no any kind of server with P2P architecture, so it is more economic. We summarize the control complexity and cost as in Table III.

TABLE III: HARDWARE COST OF DIFFERENT ARCHITECTURE

Architecture	Control Complexity and Cost
Centralized	Simple
P2P	Complex
Server-forwarding	Simple

D. Best Architecture in LAN

From the analysis above, we can see that each realization of video conference based on these three architectures has its own advantages and disadvantages. First, for centralized, it is relatively simple to control the conference and participants

and has no highly requirement on client. However, MCU needs professional device which leads to expensive cost and much post-maintenance work. Second, for P2P, without any kind of server, it is relatively economic. However, management of conference and user action is very complex and the requirement on processing ability of client need to be reconsidered. Third, for server-forwarding, central server is responsible for management conference room and participants, additionally forwarding the media stream from every client, so it requires larger network bandwidth, while require little processing ability on central server. And compared with P2P, the bandwidth and computing power needed by client is also lower.

We will choose the most appropriate one from the three kinds of architectures to implement video conference system. This system will be used in LAN for small scale team, and the feature of wanted system should include low cost, low requirements on hardware, easy to control and so on. In general, the bandwidth is sufficient within LAN. According to the above analyze, we believe that the architecture with central forwarding server is an appropriate choice. First, no professional and expensive equipment such as MCU; second, compared to P2P architecture, management is more flexible and pressure on client is smaller; Third, though central server will cost a lot of network bandwidth, should it be ok within LAN for small scale team.

IV. THE PROTOTYPE

To confirm our theoretical analysis, we will develop a video conference system with server-forwarding architecture based on JMF (Java Media Framework). And we will test the bandwidth cost by the system on server and client, and resource consumption (CPU and memory) on client. Then make a conclusion according to the test result.

JMF is a Java library that enables audio, video and other time-based media to be added to Java applications and applets [3]. This optional package, which can capture, play, stream and transcode multiple media formats extends the Java Platform, Standard Edition (Java SE) and allows development of cross-platform multimedia applications. For real time communication, JMF support Real-time Transport Protocol [4] and Real-time Control Protocol [5].

The API to implement real time application can work with the function of capturing, playing, streaming the multiple media. The media stream includes video captured by webcam and audio captured by headset, or the media stream can be obtained from saved file, though not real time. The sender processes the media stream from device or file, and then sends to network. The receiver receives media stream from the network, and it can play the media, save the media, even process the media and send it to network again. Fig. 4 and Fig. 5 show how the JMF sends and receives media stream.

We implement the video conference system exactly as Fig. 3 [6]. We create meeting room on central server, at the same time, make some configuration about the room, such as whether public or private, password, the limited number of room and so on. When a new user joins into the room, the central server will notify the information of new user to all the other participants, and forward the media stream from new user to the other participants in the room. What's more, the

central server will push the information and media stream of current participants to the new user. When a user leaves the conference, the central server will broadcast the information to the certain user, and stop sending media stream to that user.

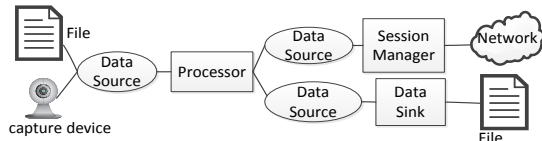


Fig. 4. Sending media stream.

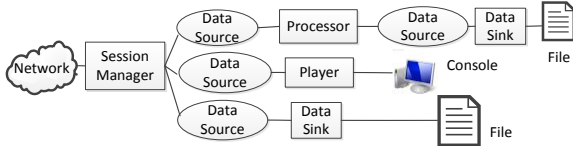


Fig. 5. Receiving media stream.

We choose H263 to encode the video. JMFF supports three level of size: level1 (128×96), level2 (176×144), level3 (352×288). For better quality, we choose the level3. In real application, we use webcam and headset to capture multimedia stream. However, due to the irregular movement of users, there exists a large jitter on resource consumption, so we use the saved media file to replace real time media, which makes it easy to verify former theory analysis.

We run the system with universal computer. Following is the feature of central server: 32-bit Windows 7 system, CPU: 3.1 GHz, memory: 4 GB.

TABLE IV: RESOURCE CONSUMPTION ON SERVER

num	cpu (%)	mem (M)	Received BW(KB/s)	Sent BW(KB/s)
1	0	18.6	48	0
2	0	19.1	96	96
3	0	19.9	144	288
4	0	20.5	192	576
5	1	21.4	240	880
6	2	22.5	288	1420

From Table IV, we can see, if the picture size is level3 (352×288), the size of composed media stream from single client is 48 Kb/s. With the expansion of conference room, there is no obvious increase on CPU and memory consumption, however, bandwidth consumption of server increase at square level; all these are close to our theoretical analysis.

Following is the feature of client: 32-bit Windows 7 system, CPU: 2.0 GHz, memory: 2 GB.

TABLE V: RESOURCE CONSUMPTION ON CLIENT

num	cpu (%)	mem (M)	Download Bandwidth	Upload Bandwidth
1	16	50	0 KB/s	48 KB/s
2	20	57.3	48 KB/s	48 KB/s
3	22	62.7	96 KB/s	48 KB/s
4	28	68.5	144 KB/s	48 KB/s
5	33	75.2	196 KB/s	48 KB/s
6	40	86.4	244 KB/s	48 KB/s

From Table V, we can see that when a new user joins into the room, the consumption of CPU will increase about 4% ~ 5% and bandwidth increases at a linear level.

Test result is consistent with our previous theoretical

analysis, proving that this is a feasible architecture. There is no problem to use the video conference system on common computer.

V. CONCLUSION

This paper intends to find the best architecture to implement video conference system in LAN. Firstly we analysis three architectures: the centralized, the P2P and the server-forwarding. By comparing the advantages and disadvantages of these three architectures and combining with our target feature – small-scale team in LAN, we think the server-forwarding is the best choice to implement the system. Then we implement the video conference system based on JMFF, and test the performance and resource consumption of the system. By analyzing the test data, we find the measured data is very close to the theoretical analysis, which confirm that server-forwarding is the most appropriate choice to implement video conference system in LAN.

ACKNOWLEDGMENT

The authors thank for the important national science & technology specific projects: Next-Generation Broadband Wireless Mobile Communications Network (2012ZX03002008-002-03), National Key Basic Research Program of China (973 program) (2011cb302506); National Natural Science Foundation Under Grant 61170274; National Key Basic Research Program of China (973 program) (2009cb320504) and the Open Project of Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory (ITD-U14003/KX142600010).

REFERENCES

- [1] Y. Z. Zou and C. J. Chen, "MCU system software in video conference network," in *Proc. 1996 International Conference on Communication Technology*, 1996, pp. 65-68.
- [2] J. S. Ma *et al.*, "Study of P2P-based video conference system," *Computer Engineering and Design*, vol. 9, pp. 2063-2065, 2007.
- [3] P. Zeng *et al.*, "A grouped network video conference system based on JMFF in collaborative design environment," in *Proc. Int. Conf. on Sig. Image Technologies and Internet-Based System*, 2007, pp. 129-136.
- [4] T. Friedman, R. Caceres, and A. Clark, "IETF RTP control protocol extended reports (RTCP XR)," 2010.
- [5] *Real Time Control Protocol (RTCP) Attribute in Session Description Protocol (SDP)*, 2010.
- [6] H. T. Li, W. S. Chen, and T. S. Li, "A Java JMFF-based architecture for presenting IP/TV multicast RTP streams," 2000.



Kai Shuang was born in Liaoning province, China, in 1977. He received his Ph.D. degree from the State Key Laboratory of Networking & Switching Technology, Beijing University of Posts & Telecommunications (BUPT) in 2006, majored in computer application technology. He is currently an associate professor with the BUPT. His research interests include cloud computing and mobile internet.

Chunhua Wu was born in Hubei province, China, in 1990. He is now a master candidate with the State Key Laboratory of Networking & Switching Technology from BUPT. His research interests include cloud computing and mobile internet.

Zhenjiang Dong is currently the vice president of the ZTE Cloud Computing & IT Research Institute. His research interests include cloud computing.