Dynamic Scalable Model for Video Conferencing (DSMVC) using Request Routing

Adeel Anwar Abbasi*, Tahir Mehmood**
{*Department of Computer Sciences, Shaheed Zulfiquar Ali Bhutto Institute of Science and Technology} (SZABIST)
Islamabad, Pakistan
Email: confusedimages@yahoo.com

Abstract: The Session Initiation Protocol (SIP) is an application-layer control protocol which is used to create, alter and cease multimedia sessions. These multimedia sessions include call and video conferences, Internet telephony and other such events. In general, SIP has centralized videoconference system, which faces the drawbacks of scalability and bandwidth. When there is a need to have a multimedia conference on a large-scale, the issue of excessive load on Conference Server (CS) must be resolved. In this paper we deal with this bottleneck issue on Conference Server by examining the shortcomings of current multimedia conference model. We propose a dynamic scalable model for SIP-based conferences DSMVC (Dynamic Scalable Model for Video Conferencing), which will help us reducing the traffic from CS. In this model, clients act as a CS rather than using a one CS for every client. As a result, when the number of clients increases, the number of CS increases accordingly. Since hosts play a role of CS in this model, they need to communicate with each other and in order to do that, a client needs to know the IP address of an appropriate networking partner. DSMVC request-routing helps a client identifying the IP address of its suitable partner. We analyze the load on the network and thus assess the scalability of the conference system. By simulation of the assessment, DSMVC conference system is found to have scalability for the number of clients.

Keywords: SIP, CS, DSMVC, DNS

1. INTRODUCTION

Since internet is being widely used over the world, its bandwidth has increased over the last decade. Moreover, communication using multimedia data, for instance Video conference system, is getting very popular in present time. Currently, there are two standards to maintain the progress of video conference systems, Session Initiation Protocol (SIP) [1] and ITU-T H.323 [2]. The structural design of SIP is based on the client-server model. There are three models for conferencing controls: Loosely coupled conference, fully distributed multiparty conferencing and tightly coupled conference [3]. All of them lack the ability to increase the capacity of conference system dynamically while a conference is in session. This paper presents a Dynamic Scalable Model for Video Conferencing (DSMVC) to surmount this limitation. In this model a host acts as a CS, its SIP client application connects with its own CS, and finally the CS connects with the CS of another client. When the number of users in a conference increases, the number of CSs also increases at that location, which in turn reduces the traffic off the server and distributes it among the clients accordingly. Since hosts work as a CS and need to connect with other clients, the IP address or host name of suitable clients is required in order to connect with them. This task is carried by DSMVC which uses request-routing method to find a suitable partner for a host. Request-routing process delivers a host’s requests to the most adequate server dynamically by utilizing the dynamic information of the request [4]. By using DNS server, DSMVC tells the new host its optimal connection partner at the name resolution. Throughout the conference, DNS server corrects the extensive information such as routing tree of the conference [4]. When a new host participates in the conference, it sends packets to create connection with the other participant, but destination address of these packets is the common host name prepared for participation [5]. The DNS server checks the client that initiates the request, determines a suitable connection partner for that client and then sends a reply to that partner’s IP address [4]. This allows a new client to know more about its connection partner. In this paper we discus existing SIP model in Chapter 2. In section 3 we highlight limitations of existing SIP model. In section 4 we proposed our new model DSMVC. In section 5 we mention some related work and at the end in section 6 we made simulation of models and simulation readings.

2. EXISTING SIP VIDEO CONFERENCE

In SIP multipoint conference, CS connects with all the clients by gathering their audio and video streams [7]. It then mixes those streams and sends the mixed stream to all the clients. This sets up a conference among the CS and all the hosts. Therefore, if the number of users in a conference increases, it results in excessive processor and network load on CS and thus the conference system becomes overwhelmed [7]. The procedure of dividing the workload off the CS must be followed in order to divert the excessive load off the CS. Currently, multiple CS are being used to overcome this bottleneck issue. The load of numerous users gets distributed among two or more CS in
order to lighten up the load from each CS. Although this method is effective to some extent, it too has some limitations. As more and more hosts join a conference, the number of servers needed to reduce the load will increase accordingly. Also, the location of these servers is stationery but the location of clients may vary dynamically. Therefore, if multiple users gather somewhere, the CS near that location will become overloaded nevertheless. Figure 1 shows the SIP video conferencing model using one main conference server. As we can see from figure conference server communicate with all host. It collects audio and video stream, after the mixing it sends to all hosts. It holds one-to-one communication with all hosts so more load on conference server; it will create bottle neck in conference server.

4. PROPOSED METHOD DSMVC (Dynamic Scalable Model for Video Conferencing)

A new multimedia conference system ought to be established which is not restricted by the number or location of the clients. To develop such a system, a procedure of load sharing must be followed which functions effectively regardless of the number or physical location of users. Since the method which is being followed currently for load sharing using multiple CS does not offer scalability option, we propose a new method called DSMVC (Dynamic Scalable Model for Video Conferencing). This model offers two features to reduce the load off the CS effectively. These are described in the sections below:

![Existing SIP Video Conferencing Model](image1.png)

**Figure 1: Existing SIP Model [8]**

3. LIMITATION OF EXISTING SIP MODEL

Existing SIP Model support one-to-one communication with all clients, and it is used for remote conference, and each host which is participating in the conference must communicate with the conference server. Each SIP client sends its audio and video stream to conference server. When conference server receives all streams from SIP client, it selects some streams, mixes them, and distributes it to each SIP client. In other words, conference server holds one-to-one conference with all SIP clients. Therefore, when the number of SIP client increases, network load on conference becomes heavier, and if many hosts participate in the same conference, conference server may become over load which create bottle neck in conference server, and conference server may stop. Then conference system is no longer operational mode. The simplest solution of this limitation is to use multiple conference servers to balance the load of mixing and distribution. The number of Conference Server and its location is unchangeable. But, the number and location of SIP clients change dynamically, so required number of conference servers which at optimal place for load sharing cannot necessarily be provided in accordance with the number and location of hosts by this method.

![Proposed SIP Video Conferencing Model](image2.png)

**Figure2: DSMVC (Dynamic Scalable Model for Video Conferencing)**

4.1. LOAD DISTRIBUTION BY SIP HOSTS

A CS is responsible for distributing streams to clients in the current model. So as the hosts in a conference increase, the load of transporting streams to users increases as well. To avoid this difficulty, DSMVC relies on hosts to send their own streams and is not dependent upon CS. According to the proposed model, a host executes the SIP application as well as the CS application which connect with each other. Afterwards, the CS application of one client connects with the CS application executed by the other clients. This ensures the availability of servers corresponding to the number and location of clients in a session.

4.2. REQUEST ROUTING THROUGH DNS SERVER

A process in which a user’s request is forwarded to the most suitable partner dynamically is called request routing. A redirection server records the static and
dynamic information of clients. When it receives a request from a host, it determines the best surrogate based on clients’ static and dynamic information and notifies the host that made the request. This allows clients to know their suitable connection partners. Domain Name System or DNS is one of the various and most efficient methods for request routing. When a request for a multimedia session is made, a user releases a common host name that is used to make a connection with other clients at destination address of the packets. This host name is received by the DNS server, which translates it to the IP address.

4.2.1. SELECTING THE BEST HOST THROUGH REQUEST ROUTING

In order for a user to participate and receive streams in a conference, he must connect to other participating hosts as they serve as a CS. Hence, request routing is very effective way to notify clients about their best connection partners. Also, since request routing using DNS is much easier to realize than several other methods, DSMVC takes on this method to connect clients in a multimedia conference.

4.3. THE POLICY FOR SELECTING CONNECTION PARTNER

To determine the best surrogate for a host, there need to be a criterion by means of which a DNS sever can choose suitable candidates for connection. The DNS server should adopt the following criteria when determining the best partner. A user whose bandwidth comes to an end should not be chosen for a connection. A user who belongs to the same organization as the one who made the request should be favoured over the one from a different organization since the connection within an organization is more efficient than the one outside it. In a situation where two hosts are under the same conditions, the one whose processor and network load is the least should be chosen. The steps, which a DNS server goes through in order to select an optimal participating surrogate for a new client, are summarized below:
1. Never select a host whose bandwidth comes to an end.
2. If possible, select a host who belongs to the same organization as the host who made a request. In case of several such hosts, select the one whose processor and network load average among all hosts is the minimum.
3. If there is no such client, select the one with the least processor and network load.

5. RELATED RESEARCH

Another approach for load sharing of conference server is distributed architecture; in which large conference include many clients situated in several domains, which are interconnected by a WAN such as the Internet. In this distributed model there are different domains: (i) an arbitrary number of Clients (UAs), (ii) a SIP Proxy Server coupled with a SIP registrar server and (iii) a conference server. The proxy server and conference server are completely distributed therefore the architecture is highly scalable without assuming the availability of multicast support. The clients are fully SIP-compatible and a point-to-point call can be directed without pass through Proxy Servers. The messaging between proxy server and conference server is also SIP-compatible. Moreover this allows the corporate sites to use their own third party conference server and other facilities available for conferencing support that is SIP-compatible.

6. EVALUATION

In this section we evaluate the performance of DSMVC by simulation. We simulate DSMVC in the ns2 simulator. The criterion for evaluation is the maximum network load; the load on the link whose load is the heaviest of all links. At simulation, the hosts which participate in conference are randomly selected from all hosts. We simulate in four following cases.

6.1. SIMULATION OF EXISTING MODEL

Existing SIP Model conference systems has one major problem, network load of the system are concentrated on conference server. So they use several conference servers to share the load. But if the number of clients increases, conference servers may become overload. Figure 3 shows the result of simulation when the number of hosts increases from 1 to 20 with one main conference server. Maximum load of existing methods increase linearly related to the number of hosts. Therefore if the number of hosts increases, conference system which uses this method may become overload.

![Figure 3: Existing Method](image-url)
6.2. SIMULATION OF EXISTING MODEL WITH TWO CONFERENCE SERVERS

There is another solution for load sharing is to use multiple conference servers. In this case we take same number of client with two conference servers. Figure 4 shows the result of simulation when the number of hosts increases from 1 to 20 with two conference servers. With the help of two conference servers, network load will decrease but it still does not have scalability. As more and more hosts join a conference, the number of servers needed to reduce the load will increase accordingly.

Figure 4: Existing Method with 2 Conference Servers

6.3. SIMULATION OF EXISTING MODEL WITH SIX CONFERENCE SERVERS

In this case we take same number of client with six conference servers. Figure 5 shows the result of simulation when the number of hosts increases from 1 to 20 with six conference servers. With the help of six conference servers load will more decrease than with two conference servers. According to this solution, more conference servers will decrease more load, but this solution also has the limitation. The location of these servers is stationary but the location of clients may vary dynamically. Therefore, if multiple users gather somewhere, the CS near that location will become overloaded nevertheless.

Figure 5: Existing Method with 6 Conference Servers

6.4. SIMULATION OF DYNAMIC SCALABLE MODEL FOR VIDEO CONFERENCING (DSMVC)

We propose new method DSMVC to provide scalability for the number of hosts. Therefore we should check if DSMVC has scalability for host or not. We simulate the relationship between the load and the number of hosts to prove the scalability of DSMVC. We measure the maximum network load of DSMVC. When the number of hosts increases from 1 to 20 gradually, the simulation will be as shown in figure 6. It shows that the network load is decreasing as clients increase at DSMVC and it distributes the network load according to the number of clients. In other words, DSMVC provides scalability for the Video conference system. Figure 6 shows clear result that Dynamic Scalable Model for Video Conferencing (DSMVC) has more scalability for clients than other existing method. We measure the maximum network load of existing methods to confirm these existing methods do not have scalability for the number of hosts. So our method Dynamic Scalable Model for Video Conferencing (DSMVC) provides best solution for scalability when number of clients increase.

Figure 6: Dynamic Scalable Model for Video Conferencing (DSMVC)

6.5. SIMULATION READING

<table>
<thead>
<tr>
<th>user</th>
<th>load for one server (Mbps)</th>
<th>load for two servers (Mbps)</th>
<th>load for six servers (Mbps)</th>
<th>load for DSMVC (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>2</td>
<td>0.5</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>3</td>
<td>1.333333333</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>5</td>
<td>1.666666667</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>6</td>
<td>2.333333333</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td>7</td>
<td>2.666666667</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td>8</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>9</td>
<td>3.333333333</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>18</td>
<td>10</td>
<td>3.666666667</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>11</td>
<td>4.000000000</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>22</td>
<td>12</td>
<td>4.333333333</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>13</td>
<td>4.666666667</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>26</td>
<td>14</td>
<td>5.000000000</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>28</td>
<td>15</td>
<td>5.333333333</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>30</td>
<td>16</td>
<td>5.666666667</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>32</td>
<td>17</td>
<td>6.000000000</td>
<td>2</td>
</tr>
<tr>
<td>17</td>
<td>34</td>
<td>18</td>
<td>6.333333333</td>
<td>2</td>
</tr>
<tr>
<td>18</td>
<td>36</td>
<td>19</td>
<td>6.666666667</td>
<td>2</td>
</tr>
<tr>
<td>19</td>
<td>38</td>
<td>20</td>
<td>7.000000000</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>40</td>
<td>20</td>
<td>7.333333333</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1: Network Load Using Various Methods of Load Sharing
7. CONCLUSION AND FUTURE WORK

In this paper, we propose new method DSMVC that adds the functionality of Conference Servers to the SIP clients, use these clients instead of Conference Servers, and adapt CDN Request-Routing for scalable conference system to for load sharing. And, we prove the efficiency of DSMVC by simulation. Now DSMVC is helpful in one domain to distribute the streams, and it does not support the hosts of different domains participating in Video Conferencing. Some of our future works include improving the DSMVC’s performance of load sharing in multiple domains to provide larger video conferencing in SIP network.

REFERENCES