Yue Lu, Yong Zhao, Fernando Kuipers, and Piet Van Mieghe

Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Neth
{Y.Lu,F.A.Kuipers,P.F.A.VanMieghem}@tudelft.nl

Abstract. More and more free multi-party video conferencing appli
tions are readily available over the Internet and both Server-to-Cli
(S/C) or Peer-to-Peer (P2P) technologies are used. Investigating th
mechanisms, analyzing their system performance, and measuring th
quality are important objectives for researchers, developers and end us
In this paper, we take four representative video conferencing appli
tions and reveal their characteristics and different aspects of Quality
Experience. Based on our observations and analysis, we recommend
incorporate the following aspects when designing video conferencing
plications: 1) Traffic load control/balancing algorithms to better use
limited bandwidth resources and to have a stable conversation; 2)
traffic shaping policy or adaptively re-encode streams in real time
limit the overall traffic.

This work is, to our knowledge, the first measurement work to stu
and compare mechanisms and performance of existing *free multi-pa*
video conferencing systems.

1 Introduction

The demand for video conferencing (VC) via the Internet is growing
services are provided in two different ways: (1) either utilizing a hig
VC room system with professional equipment and dedicated bandwid
implementing a VC application on personal computers. The first cate
guarantee quality, but it is costly and limited to a fixed location, while t
category is often free of charge and easy to install and use, although th
cannot be guaranteed.

In this paper, we focus on studying *free* applications that provide mu
($\geq$ 3 users) VC on the Internet, and focus on the following questions:

- *How do multi-party VC applications work?*
- *How much resources do they need?*
- *What is the Quality of Experience (QoE)?*
- *What is the bottleneck in providing multi-party VC over the Intern*
- *Which technology and architecture offer the best QoE?*

M. Crovella et al. (Eds.): NETWORKING 2010, LNCS 6091, pp. 96–108, 2010.

The remainder of this paper is organized as follows. Section 2 presen
work. In Section 3, eighteen popular VC applications will be introd
classified. Section 4 describes our QoE measurement scenario. Section
will show the measurement results obtained. Finally, we conclude in S

2 Related Work

Most research focuses on designing the network architectures, mechan
streaming technologies for VC. In this section we only discuss the work
ing and comparing the mechanisms and performance of streaming app

Skype supports multi-party audio conferencing and 2-party video ch
and Schulzrinne [1] analyzed key Skype functions such as login, call
ment, media transfer and audio conferencing and showed that Skyp
centralized P2P network to support audio conferencing service. Cicco
measured Skype video responsiveness to bandwidth variations. Their r
dicated that Skype video calls require a minimum of 40 kbps available b
to start and are able to use as much as 450 kbps. A video flow is m
tic through congestion control and an adaptive codec within that b
interval.

Microsoft Office Live Meeting (Professional User License) uses a S
tecture and has the ability to schedule and manage meetings with up
participants. However, only few participants can be presenters who ca
their videos and the others are non-active attendees.

Spiers and Ventura [3] implemented IP multimedia subsystem (IM
VC systems with two different architectures, S/C and P2P, and measu
signaling and data traffic overhead. The results show that S/C offe
network control together with a reduction in signaling and media
whereas P2P allows flexibility, but at the expense of higher overhead.

Silver [4] discussed that applications built on top of web browsers
the world of Internet applications today, but are fundamentally fla
problems listed include delays and discontinuities, confusion and error
interfacing and limited funtionality.

Trueb and Lammers [5] analyzed the codec performance and securi
They tested High Definition (HD) VC and Standard Definition (SD)
characteristics and their corresponding video quality. In their results,
vides a better video quality at good and acceptable network conditions
poor network conditions HD and SD have similar performance.

Few articles compare the different types of existing free multi-party V
or measure their QoE. In this paper, our aim is to provide such a compa

[1] http://www.mebeam.com/
[2] http://www.qnext.com/
[3] http://www.vsee.com/
[4] http://www.nefsis.com/leads/free-trial.aspx

frame rate they can support (the best video quality they can provide), the maximum number of simultaneous conference participants, and the categ
or P2P) they belong to in Table 1.

Table 1. Popular video conferencing applications

	Max. frame rate (frames/second)	Max. # of simultaneous video participants	S/C or
Eedo WebClass		6	web-base
IOMeeting	30	10	web-base
EarthLink	30	24	S/C
VideoLive	30	6	web-base
Himeeting	17	20	S/C
VidSoft	30	10	S/C
MegaMeeting	30	16	web-base
Smartmeeting	15	4	S/C
Webconference	15	10	web-base
Mebeam		16	web-base
Confest	30	15	S/C
CloudMeeting	30	6	S/C
Linktivity WebDemo	30	6	web-base
WebEx	30	6	web-base
Nefsis Free Trial	30	10	S/C
Lava-Lava	15	5	decentraliz
Qnext		4	centralize
Vsee	30	8	decentraliz

Even though there exist many free VC applications, many of them
to be instable once installed. From Table 1, we observe that the maxim
rate is 30 frames/s which corresponds to regular TV quality. All ap
support only a very limited number of participants and the applicat
support more than 10 simultaneous participants all use a centralized
work structure.

Many other popular online chatting applications (like Skype, MS
messenger, Google talk, etc.) only support multi-party audio confer
2-party video conference, and therefore are not considered here.

4 Experiments Set-Up

We have chosen four representative applications to study:

- *Mebeam*: web-browser based S/C with a single server center.
- *Qnext (version 4.0.0.46)*: centralized P2P. The node which hosts th
  is the super node.

the four architectures under which all eighteen applications in Table
classified.

We have performed two types of experiments: (1) local lab experime
posed of standard personal computers participating in a local video co
in order to investigate the login and call establishment process, as w
protocol and packet distribution of the four VC applications; (2) glob
ments, to learn more about the network topology, traffic load and QoE
more realistic international video conference is carried out.

The global measurements were conducted during weekdays of May,
der similar and stable conditions[5]:

- Client 1: 145.94.40.113; TUDelft, the Netherlands; 10/100 FastEth
  Client 2: 131.180.41.29; Delft, the Netherlands; 10/100 FastEthern
  Client 3: 159.226.43.49; Beijing, China; 10/100 FastEthernet;
  Client 4: 124.228.71.177; Hengyang, China; ADSL 1Mbit/s.
- Client 1 always launches the video conference (as the host);
- Clients 1, 3 and 4 are behind a NAT.

To retrieve results, we used the following applications at each particip

- *Jperf* to monitor the end-to-end available bandwidth during the w
  cess of each experiment. We observed that usually the network is qu
  and that the available end-to-end bandwidth is large enough for dif
  plications and different participants.
- *e2eSoftVcam* to stream a stored video via a virtual camera at
  participant. Each virtual camera is broadcasting in a loop a "Nev
  (.avi file) with a bit rate of 910 Kbit/s, frame rate of 25 frames/s
  480x270;
- *Camtasia Studio 6*. Because all applications use an embedded media
  display the Webcamera streaming content, we have to use a screen
  to capture the streaming content. The best software available t
  Camtasia, which could only capture at 10 frames/s. In order to h
  comparison of the original video to the received video, we captured
  the streaming videos from all participants, but also the original s
  video from the local virtual camera[6].
- *Wireshark* to collect the total traffic at each participant.

[5] We have repeated the measurements in July, 2009 and obtained similar
those obtained in May 2009.

[6] We assess the video quality using the full reference batch video quali
(bVQM) which computes the quality difference of two videos. Captur
frames/s a video with frame rate of 25 frames/s may lead to a different
bVQM score. However, because the video used has a stable content (ther
small changes in the person profile and background), we do not expect a
viation in bVQM score with that of the 25 frames/s video. The results ar
for 10 frames/s videos.

Mebeam: We open the *Mebeam* official website to build a web video-c
and all participants enter the room. The traces collected with *Wireshark*
that two computers located in the US with IP addresses 66.63.191.20
Server) and 66.63.191.211 (Conference Server) are the servers of *Mebe*
client first sends a request to the login server, and after getting a resp
up a connection with the single conferencing server center. When the c
host leaves from the conference room, the meeting can still continue.
uses TCP to transfer the signals, and RTMP[7] to transfer video and au

Qnext: The data captured by *Wireshark* reveals two login severs located i
Each client first sends packets to the login servers to join the network. Aft
a response, they use SSLv3 to set up a connection with the login servers.
establishment process, each client communicates encrypted handshake
with 3 signaling servers located in the US and Romania and then uses SS
up a connection between the client and the signaling server. When client
another client B to have a video conference and client B accepts A's requ
use UDP to transfer media data between each other. In a conference, the
one host and other clients can only communicate with the host. The h
super node in the network. When the host leaves the meeting, the me
end. If another node leaves, the meeting will not be affected. *Qnext* uses
signaling and UDP for video communication among participants.

Vsee: Each client uses UDP and TCP to communicate with the web
login process. In the call establishment process, after receiving the inv
the host, each client uses[8] T.38 to communicate with each other. *Vsee*
web servers: during our experiment, one in the Netherlands, one in Ca
7 located in the US. *Vsee* has a full-meshed P2P topology for video
However, only the host can invite other clients to participant in the co
When the host leaves the meeting, the meeting cannot continue. Other
leave without disrupting the meeting. *Vsee* is a video-conferencing and
collaboration service. The communication among users is usually of the
using UDP, with automatic tunneling through a relay if a direct con
not available.

Nefsis: In the login process, the clients first use TCP and HTTP to c
the Virtual Conference Servers (with IP addresses 128.121.149.212 in th
118.100.76.89 in Malaysia) and receive information about 5 other acce
from the Virtual Conference Servers. These 5 access points are also
server centers owned by *Nefsis*, and they are located in the Netherlands
dam and Amsterdam), in the UK, India, Australia, and Singapore. Af

[7] Real-Time Messaging Protocol (RTMP) is a protocol for streaming audio,
data over the Internet, between a Flash player and a server.

[8] T.38 is an ITU recommendation for fax transmission over IP networks in

set-up an end-to-end connection to communicate with each other direct uses TCP for signaling and delivering streaming data.

5.2 Packet Size Distribution and Traffic Load

To differentiate between non-data packets, video and audio packets formed three local experiments for each application. The first experin two computers with cameras and microphones to have a video confe the second experiment, two computers are only equipped with microph without cameras (no video packets will be received). In the third exp two computers set-up a connection, both without microphones and ca only non-data packets will be exchanged).

Based on *Wireshark* traces, we could distill for each VC application t size range as shown in Table 2:

Table 2. The packet size distribution of *Mebeam*, *Qnext*, *Vsee* and *N*

Packet size	*Mebeam*	*Qnext*	*Vsee*	*Ne*
Audio packet	$>$ 50 bytes	72 bytes	100 $\sim$ 200 bytes	100 $\sim$ 20
Video packet	$>$ 200 bytes	50 $\sim$ 1100 bytes	500 $\sim$ 600 bytes	1000 $\sim$ 1
Signaling packet	50 $\sim$ 200 bytes	50 $\sim$ 400 bytes	50 $\sim$ 100 bytes	50 $\sim$ 100

Other interesting observations are: 1) If the person profile or backgrou in the camera change/move acutely, a traffic peak is observed in our The traffic does not necessarily increase as more users join the conferen shows the change of the average traffic load at each user when a new pa joins the conference[9]. The decreasing slope after 3 users indicates that *Qnext* and *Vsee* either re-encoded the videos or used traffic shaping in reduce/control the overall traffic load in the system. We can see from F only the traffic load at *Nefsis* clients does not decrease when the numbe conferencing participants reaches to 4. Therefore, we introduced more pa into the video conference for *Nefsis*, and we found that the traffic at ea user starts to decrease at 5 participants. Hence, we believe that in order t more simultaneous conference participants, the overall traffic has to be c

Fig. 1 illustrates that, compared with the traffic generated by *Nefsis* w the same coding technology and the same frame rate on the same vid and *Vsee* generate most traffic, especially the host client of *Qnext*. This i *Qnext* and *Vsee* use P2P architectures where the signaling traffic overhea more than the traffic generated by a S/C network with the same number ipants. The host client (super node) of *Qnext* generates 3 times more tr

[9] We captured the packets after the meeting was set up and became stable.

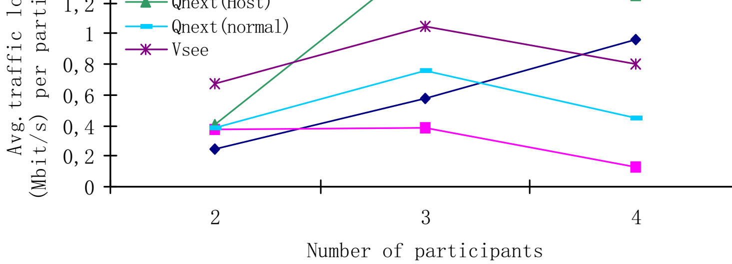


Fig. 1. The average traffic load at an end-user when the number of conferen
pants increases from 2 to 4 (*Qnext* is limited to 4 participants)

other normal clients. Hence, for this architecture, a super-node selectio
recommended to choose a suitable peer (with more resources, for examp
super node.

Fig. 1 also shows that *Mebeam* generates least traffic. Considering that
all traffic load, which can be supported in a VC network, has an upperbou
the limited users' bandwidth, and each *Mebeam* client generates much l
than the three other applications, it clarifies why *Mebeam* can support
taneous video users while *Nefsis* can only support 10 users, *Vsee* can s
users and *Qnext* can support 4 users.

5.3 Quality of Experience (QoE)

QoE can be measured through objective and subjective measurements. I
tion, we assess via global measurements the QoE at the end user with r
their video quality, audio-video synchronization, and level of interactio

Video Quality. In the objective measurements, we use bVQM (Ba
Quality Metric) to analyze the VC's video quality off-line. bVQM takes
nal video and the received video and produces quality scores that refle
dicted fidelity of the impaired video with reference to its undistorted co
The sampled video needs to be calibrated. The calibration consists of e
and correcting the spatial and temporal shift of the processed video sequ
respect to the original video sequence. The final score is computed us
ear combination of parameters that describe perceptual changes in vid
by comparing features extracted from the processed video with those
from the original video. The bVQM score scales from 0 to approximatel
smaller the score, the better the video quality.

[10] According to [7], bVQM scores may occasionally exceed 1 for video scen
extremely distorted.

Table 3 provides the bVQM scores for VC service per participant.

Table 3. The video quality of *Mebeam*, *Qnext*, *Vsee* and *Nefsis* at 4 cli

VQM score	Client 1	Client 2	Client 3	Client 4	Avera
Mebeam (Flash video, MPEG-4)	0.63	0.41	0.94	0.86	0.71
Qnext (MPEG-4, H.263, H.261)	1.05	0.94	0.63	0.83	0.86
Vsee	0.78	0.82	0.80	0.79	0.80
Nefsis (MPEG-4, H.263, H.263+)	0.34	0.61	0.61	0.87	0.61

Table 3 indicates that *Nefsis* features the best video quality among plications, although with an average bVQM score of 0.61 (its quality is or which will be explained later with the subjective measurements). The higl score (the worst video quality) appears at Client 1 (the super node) of *Qn* erally speaking, all four VC applications do not provide good quality[12].

Because no standard has been set for what level of bVQM score corre what level of perceived quality of a VC service, we have also conducted s measurements. We use the average Mean Opinion Score (MOS) [8], a me user perceived quality, defined on a five-point scale[13]: 5 = *excellent*, 4 = *fair*, 2 = *poor*, 1 = *bad*.

We gave 7 different quality videos generated by VC applications to 2 who gave a subjective MOS score independently. We also objectively c their bVQM scores. Fig. 2 shows the correlation between the objectiv scores and the subjective MOS values.

We mapped between the bVQM scores and the average MOS score persons, and found that they have a linear correlation in the range 0.3 score$\leq$ 1. Hence, the VC's video quality is predictable when using the metric bVQM.

Compared with the video quality of a global P2PTV distribution servi has an average MOS value of 4 [9], the video quality of a global VC servi (with an average bVQM score of 0.74 and MOS value of around 2.2), be VC service requires end users to encode and upload their streams in

[11] *VirtualDub* is a video capture and video processing utility for Microsoft W

[12] We also objectively measured the audio quality using metric PESQ-LQ (F Evaluation of Speech Quality-Listening Quality) [6] [8] and found that the average score (scale from 1.0 to 4.5, where 4.5 represents an excellent audi is 2.24, 2.68, 3.08 and 3.15 for *Mebeam*, *Qnext*, *Vsee,* and *Nefsis,* respectiv

[13] The threshold for acceptable TV quality corresponds to the MOS value 3.

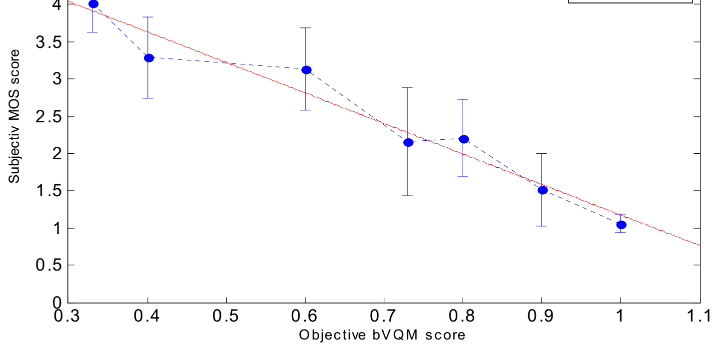


Fig. 2. Relation between bVQM and MOS for video conferencing serv

Even the local uploaded video has a largely degraded quality althoug the best among all participants.

Audio-Video Synchronization. The relative timing of sound and in tions of a streaming content may not be synchronized.

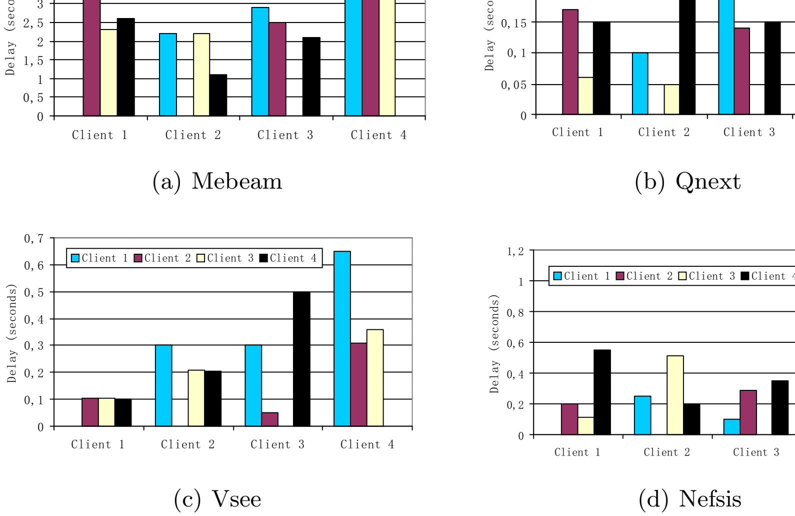
ITU [10] [11] has suggested that the viewer detection thresholds of au lag are about $+45$ ms to -125 ms, and the acceptance thresholds are a ms to -185 ms, for video broadcasting.

To analyze the A/V synchronization provided by each VC applicatio an "artificially generated" video test sample, in which the video and au forms are temporally synchronized with markers. Similar to the exper testing the video quality, we captured at each end user the videos from participants. When the audio and video tracks were extracted and com line, there was an average difference in time between the two tracks of ms for *Mebeam*, 470 ms for *Qnext*, 400 ms for *Vsee* and 350 ms for *Ne* large audio-video lags are mainly caused by a large amount of frame los lead to the low video quality mentioned already in Section 5.3.

Interactivity (communication delay). During a video conference it ing to have large communication delay[14]. Large communication delay im of real-time interactivity in our global multi-party VC experiments. We the video delays among participants by injecting in the network anothe video that mainly reproduced a timer with millisecond granularity.

In the video conference, this artificial "timer" video was uploaded via t camera and transmitted among the participants via the different VC app

[14] In IP video conferencing scenarios, the maximum communication delay reco by ITU is 400 ms [12].



(a) Mebeam

(b) Qnext

(c) Vsee

(d) Nefsis

Fig. 3. The video delay between different participants

At each participant, we used a standard universal Internet time as referen
displayed the "timer" videos of all participants in real time. After a 1-mi
stable video conference, we cut the captured content at each participant
tualDub to compare the "timers" between any 2 participants. For each ap
we took samples at 2 different times to calculate an average delay.

The video delays among participants are shown in Fig. 3. The x axis s
4 different clients. The y axis shows the video transmission delay from th
ipant on the x axis to the participant shown in the legend.

Fig. 3 shows that *Qnext* provides a video that is most synchronized a
clients. *Qnext*, *Vsee*, and *Nefsis* have a comparable level of average video
spectively 0.15 s, 0.27 s, and 0.41 s. However, *Mebeam* clients suffer a huge
lay (2.77 s on average), because the processing time at the serv
long.

We also measured the audio delays among participants by injecting in
an artificial DTMF (Dual-tone multi-frequency) tone. We sent and rec
audio at Client 1. Other participants kept their speaker and micropho
did not produce extra audio. Based on the recorded audio tracks, we com
time the audio marker was sent from Client 1 and the time the same aud
was heard again at Client 1 after the transmitted audio was played, reco
retransmitted by a client. The time difference is approximately twice the
audio delay plus the processing delay at a client. Our results revealed

[15] http://www.time.gov/timezone.cgi?Eastern/d/-5/java

to-end delay including the transmission delay and the delay introduced b
plication. In our experiment, the video delay represents the delay of a sa
scene that was captured at the application interfaces of the sender an
ceiver, which does not include the time used for uploading the video to t
via applications. Hence, considering the audio delay, video delay, and the
chronization discussed in Section 5.3, we can conclude that the delay in
by the application, when uploading, is large for *Qnext*.

6 Worst-Case Study

In another set of global experiments in June, 2009, our *Jperf* plots indic
the end-to-end connections of clients 3 and 4 with the host were very uns
found that the two participants in China always passively disconnected
conference or could not even log into *Mebeam*, *Nefsis* and *Qnext*. *Vsee* c
work, but the quality was awful, with bVQM scores close to 1.

In order to investigate the minimum bandwidth to support a video co
we repeated many experiments adjusting the upload rate upper-bound
Netlimiter) at each participant for a particular VC application to test
upload bandwidth minimally required to launch a video conference.

For *Qnext*, the threshold is 50 Kbit/s. If an end user's available uplo
width is $<$ 50 Kbit/s, (s)he cannot launch *Qnext*. For *Vsee*, the thresh
Kbit/s; for *Nefsis* it is 28 Kbit/s; and for *Mebeam* it is 5 Kbit/s, which w
are the minimally supported streaming bit rates set by the applications

7 Summary and Conclusions

Through a series of local and global experiments with four representat
conferencing systems, we examined their behavior and mechanisms, an
gated their login process, the call establishment process, the packet si
bution, transfer protocols, traffic load, delivery topology, and different a
Quality of Experience.

Our main conclusions from the measurement results on the traffic ch
tics of four different video conferencing systems are: (1) The QoE of mu
video conferencing is very sensitive to bandwidth fluctuations, especially
link. Hence, an adaptive bit rate/frame rate policy should be deployed;
the number of participants increases, the traffic load at each participant
always increase correspondingly (see Fig. 1), suggesting that re-encodi
video or a traffic shaping policy take place to control the overall traf
system.

[16] *NetLimiter* is an Internet traffic control and monitoring tool designed f
download/upload transfer rate limits for applications.

video and audio quality, large audio-video lag, and long communicatio some cases); (2) Only a limited number of multimedia participants are s and rare high definition webcamera streaming is supported due to th available bandwidth or the limited processing capability; (3) The existin are not reliable in the worst cases. When the network is unstable or the upload bandwidth is very limited (thresholds have been found), none of cations work properly.

It seems that the Server-to-Client architecture with many servers l over the world is currently the best architecture for providing video co via the Internet, because it introduces the least congestion at both se clients. Load balancing and load control algorithms help the overall pe of the system and the codec used is important for the quality that end ceive. The bottleneck to support video conferencing with more partici high definition streams is the overhead traffic generated by them. To sup simultaneous participants in a single conferencing session, the traffic load controlled/limited by using traffic shaping policy or re-encoding the vide

We have chosen four representative video conferencing systems for but the measurement methodologies mentioned in this paper can also b to other video conferencing applications, which could be compared with in the future.

Acknowledgements

We would like to thank Rob Kooij for a fruitful discussion on measur delay. This work has been supported by TRANS (http://www.trans-res

References

1. Baset, S.A., Schulzrinne, H.: An Analysis of the Skype Peer-to-Peer Inte phony Protocol. In: INFOCOM '06, Barcelona, Spain (April 2006)
2. De Cicco, L., Mascolo, S., Palmisano, V.: Skype Video Responsiveness to B Variations. In: NOSSDAV '08, Braunschweig, Germany (May 2008)
3. Spiers, R., Ventura, N.: An Evaluation of Architectures for IMS Based V ferencing, Technical Report of University of Cape Town (2008)
4. Silver, M.S.: Browser-based applications: popular but flawed? Informatio and E-Business Management 4(4) (October 2006)
5. Trueb, G., Lammers, S., Calyam, P.: High Definition Videoconferencing: C formance, Security, and Collaboration Tools, REU Report, Ohio Supe Center, USA (2007)
6. Rix, A.W.: A new PESQ-LQ scale to assist comparison between P.862 P and subjective MOS, ITU-T SG12 COM12-D86 (May 2003)
7. Pinson, M.H., Wolf, S.: A New Standardized Method for Objectively Video Quality. IEEE Transactions on Broadcasting 50(3), 312–322 (2004

Experience of SopCast. International Journal of Internet Protocol Techno
11–23 (2009)
10. ITU BT.1359-1, Relative timing of sound and vision for broadcasting (19
11. Lias, J.L.: HDMI's Lip Sync and audio-video synchronization for broa
home video, Simplay Labs, LLC (August 2008)
12. Bartoli, I., Iacovoni, G., Ubaldi, F.: A synchronization control scheme for
ferencing services. Journal of multimedia 2(4) (August 2007)