

Supporting E-meetings on Java capable Mobile Phones

Roland Parviainen

Roland.Parviainen@cdt.luth.se

Peter Parnes

Peter.Parnes@cdt.luth.se

Department of Computer Science and Electrical Engineering
Division of Media Technology
Luleå University of Technology
97187 Luleå, Sweden

ABSTRACT

In this paper we discuss different methods for providing group awareness to participants of e-meeting software using small, limited devices connected with limited networks such as a mobile phone. Currently small devices such as mobile phones are getting more advanced and capable of running custom software such as small Java applications while at the same time the available wireless networks are getting more pervasive and faster making it possible to connect to the Internet from anywhere, from almost any device. We describe the limitations of our target devices and methods to overcome these limitations for each media and present an implementation of some of these methods using an existing commercial e-meeting system.

Categories and Subject Descriptors

H.4.3 [Information Systems Applications]: Communications Applications

Keywords

E-meetings, Mobile and wireless collaboration, Multimedia applications, Multimedia systems

1. INTRODUCTION

Today wireless technologies such as GPRS, new 3G networks and different WLAN technologies are rapidly being developed and deployed providing wireless access to the Internet in a much larger scale than before. At the same time, new mobile platforms like PDAs, advanced mobile phones and wearable computers are becoming more and more common.

One common device that is becoming more powerful both in regards to networking support and to physical features such as CPU and memory size and is mobile phones. More and more phones that are shipped today support both custom Java applications and 2.5G or 3G technologies.

E-meetings are currently used for example for meetings, presentations and to continuously support for collaboration in a group

setting, providing among other features text messaging, audio and video, and shared collaboration tools. While providing full functionality of the e-meeting system to a mobile user would be best solution, enough features to give a sense of group awareness and an ability to be part of the work space from anywhere could be highly beneficial as well. A mobile phone, that almost every user keep with them at any time anyhow, would be the ideal platform to support this.

In this paper we focus on giving users with Java capable mobile phones the maximum amount of group awareness that the limited capabilities of the network connection and the device can support. We especially consider currently available devices: namely mobile phones that are capable of running J2ME[1] MIDP 1.0[13] Midlets, i.e. small Java applications. These devices are among the most limited that still can connect to the Internet and support custom applications. With these devices it is not generally possible to use existing tools such as mobile IP, unicast tunneling, media gateways or media transcoding as all solutions today requires more support from the device itself either in terms of processing power, etc. or from the supported networking technologies. The main research problems to solve is how to overcome the limitations of target devices and if we can add features currently unavailable in e-meeting systems that can replace media that is impossible to support or enhance the group awareness provided to the user.

By extending an existing mobile communication system, Mobile Instant Messaging, MIM[11], we show how we can support the different media of an e-meeting systems on mobile phones available to consumers today; in addition, we also add activity indicators that shows the the recent activity of any user in the e-meeting session in order to give further information to the user which is not available on the mobile device due to the different limitations.

The reminder of the paper is organized as follows. In the next sections we introduce the Marratech Pro and MIM systems, then in section 3 we describe the different limitations of the target devices. In section 3 the limitations of current Java phones are described, and in section 4 related work is presented. In sections 5 and 6 the architecture and implementation is presented. Future work is discussed in section 7. Finally, we end with conclusions in sections 8.

2. BACKGROUND

At Luleå University of Technology collaborative workspaces are used daily in a wide variety of situations. Example uses include allowing students to view classroom lectures from home, enabling members of discussion groups to interact from a distance, and pro-

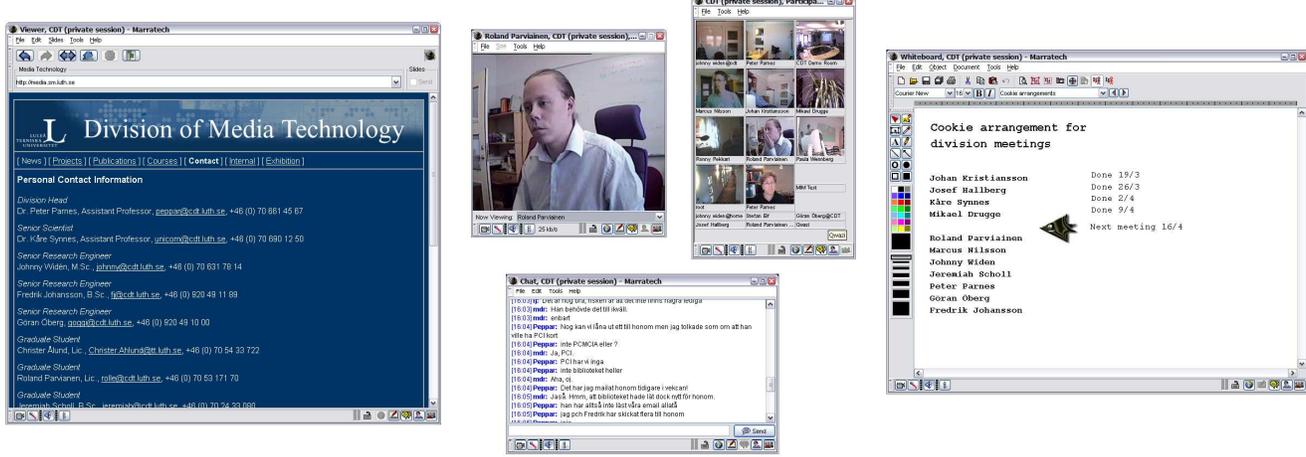


Figure 1: Screen-shot of the different tools of Marratech Pro

viding members of projects and research divisions with increased presence of each other throughout their work day. The last mentioned case is referred to as the “e-corridor”.

In the e-corridor members of the session can be active or passive and possible uses of the workspace range from giving or listening in a formal presentation to passive monitoring of video. Communicating with colleagues through the e-corridor often replace other communication media such as e-mail, phone calls or personal visits. Users with broadband Internet access at home can choose to be part of the e-corridor, join meetings and follow presentations.

2.1 Marratech Pro

Marratech Pro provides the users with the possibility to send and receive audio and video to and from other participants. In addition to audio, video and text messaging there is a shared whiteboard and a shared web browser. The chat, shared whiteboard and shared web browser tools all use proprietary protocols while the audio and video tools use standard protocols such as RTP as far as possible. It is designed to use either IP multicast or unicast. In the latter case traffic is tunneled through a media gateway called the “E-Meeting Portal”.

The video streams from participants in an e-meeting is presented “participants” window, which gives a thumbnail overview of all the video streams currently received from the group, while a “focus” window displays the video obtained from a single group member with a higher resolution and frame rate. All audio streams that are not manually muted by the user are mixed and played synchronized to the video streams. The shared whiteboard is mainly used to make quick drawings, sharing applications by incorporating screen shots and sharing Microsoft Powerpoint slides during presentations.

2.2 Mobile Instant Messaging

The Mobile Instant Messaging, MIM, system uses a hierarchical system with a MIM server running on a Home PC, which is for example the user’s actual home PC, office PC or a central network server. Different IM systems are connected by MIM plug-ins, providing the MIM server with contact lists, users’ status, functions for setting our own status, receiving and sending messages etc.

Client applications connect to the MIM server, which handles all

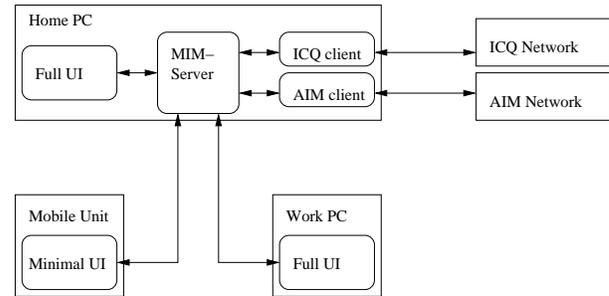


Figure 2: MIM Architecture

synchronization and stores state such as received and unread messages, history and status. When a client first connects to the server, the full state is retrieved. Further updates to the state, such as an incoming messages, are sent to all connected clients. The MIM server is always connected to the different IM systems so a user will be able to receive messages at any time, even if there is no network connectivity at the user’s current location. If no clients are connected over a duration of time, the status and auto response message can be set to pre-configured values.

If clients can not keep a connection to the MIM server open continuously, a client can also run in a polling mode, where the client regularly connects to the server for synchronization: messages that have been written are sent to the server and on to their recipients, new messages are downloaded, etc. This mode is primarily for clients running on phones, as for example the MIDP profile as used by many “Java” phones only guarantees network connectivity through outgoing HTTP requests.

See figure 2 for an example of this architecture with two different MIM plug-ins, one for the ICQ network and one for the AIM network. Three clients are connected to the MIM server: One on a PC, one on a PDA and one on a mobile phone. The clients on the PC and the PDA use a full graphical user interface, while the Phone client use a limited user interface adapted to the possibilities of the unit.

While MIM works fine with text messaging it is not always enough for interfacing with an e-meeting session. There is no way of seeing who currently is in their office, if they're in a meeting or their latest activity. Some of these problems are not unique for a user using a mobile client, but due to the limited options available the problems becomes more severe.

3. LIMITATIONS OF TARGET DEVICES

Since the main focus is on devices and network technologies available to consumers today, J2ME MIDP 1.0[13] is the obvious choice as a platform. A recent report by Zelos Group[17] states that global shipments of Java handsets will reach 86.4 million during 2003. For example, around one quarter of all Nokia handsets shipped during 2003 will support Java. The list of J2ME devices from Sun Microsystems¹ lists more than 100 J2ME devices, from 17 different manufacturers.

For some of the devices it is also possible for developers to create applications using other platforms than J2ME, such as designing applications to run directly on top of the Symbian OS on for example certain Nokia or Sony Ericsson phones. These platforms might have fewer limitations than the J2ME platform, but on the other hand the applications become much more platform dependent. Most device manufacturers also adds extra, proprietary Java APIs extending the required API in MIDP; developing for these APIs have the same problem of platform dependency.

The limitations of any solution for our target devices consist of several different parts: the limitations of the APIs provided by the standard that are available to an application developer, of the network technology used by service provider and of the physical limitations of the devices such as display size and CPU power.

In the MIDP 1.0 specification the minimum required display for a device is a screen size of 96x54 pixels with a display depth of 1 bit. Memory resources are also very limited; the MIDP 1.0 specification requires 32 kilobytes of volatile memory for the Java run-time. The only networking capabilities that a MIDP 1.0 implementation must support are through HTTP 1.1; in particular the HEAD, GET and POST requests must be supported. None of the standard network technology solutions for mobility and group communication such as IP multicast, media and transcoding gateway or mobile IP works.

Although some phones have digital still image or video cameras there are no standard API either for accessing the capture device directly or for accessing the images after they are captured, making it very cumbersome to support any image input to a Java program. While image capture is not a technology commonly associated with mobile phones, audio capture is of course one of the main points of any phone. Still, there exist no standard API for accessing the microphone or the speaker. This leaves the keypad as the only choice for input.

In Sweden, data transmission over the mobile phone network is supported over either GSM data, GPRS or UMTS. GPRS have theoretical maximum data transmission speed of 171.2 kbps requires all 8 available time slots. The number of timeslots available to a GPRS terminal is limited both by limitations in the terminals and in the network. Typically 1, 2 or 3 timeslots are available. UMTS, a 3G solution, supports data transmission speeds from 64 and 144

¹<http://wireless.java.sun.com/device/>

kbps in large macro cells, 384 kbps in micro-cells and small macro-cells and up to 2 Mbps in pico-cells. In other countries several other technologies exist such as Cdma2000 in the US, Korea and Japan and FOMA, a 3G W-CDMA service in Japan.

For our system, GPRS was chosen as the network technology since UMTS services is not yet rolled out in northern Sweden and there is still a lack of 3G terminals. Local tests in Sweden with GPRS services from several mobile phone operators have shown that download speeds above 25 kbps is almost never achieved.

3.1 Effects of the limitations

It is necessary to support unicast connections from the mobile devices as well as connections with low bandwidth and high latency. The other communication end-point need to be known by the mobile device since incoming connections might not be supported. Since HTTP requests are the only guaranteed communication media any traffic need to be transported over HTTP. Disconnections can happen often and need to be handled.

The small physical displays means that high resolution images can not easily be displayed without the need for scrolling; combined with the lack of memory it might even be impossible to scroll a image larger than the display without requesting more information from network service if the full image do not fit in the memory available to the application. This can also severely limit the number of images that can be stored or cached at the mobile device. It is mainly the shared whiteboard and video media that is affected because of this due to the large size and amount of details of whiteboard pages and the continuous nature of video.

The complete lack of audio support of course means that any direct support for audio is impossible. Input is also difficult since it is limited to what the keypad can provide.

3.2 Future standards

The Mobile Information Device Profile (MIDP) 2.0[14] includes several enhancements that are interesting from a multimedia point of view.

In addition to the already supported HTTP protocol, the only new required networking protocol is HTTPS. Support for UDP and TCP sockets are not required, but the specification recommends their implementation.

The MIDP 2.0 specification includes a subset of the Mobile Media API. It is upwardly compatible with the full API. Only audio playback (and possibly recording) is supported. No video-specific control interfaces are included. Implementations must be able to play WAV files and are free to support additional audio formats.

In addition to what is included in MIDP 2.0, the Mobile Media API (MMAPI) 1.0[15] support custom data sources through user implemented protocols, video control interfaces, and support for taking snapshots using a camera.

4. RELATED WORK

Trying to solve network heterogeneity through active media processing in media gateways is a common approach, see for example [2][10] and [16]. Another example is [7], where a multimedia gateway is used to access multicast video conferences from a web page. Several gateways for using phone to call into a ongoing video conference also exist.

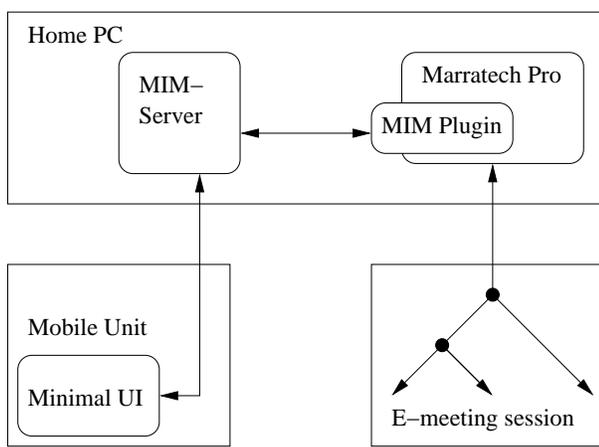


Figure 3: Overall system architecture. A MIM plugin inside the Marratech Pro application communicates with the MIM system while the Marratech Pro application communicates with the E-meeting session over IP Multicast

The ideas of activity indicators as a mean for providing group awareness have been explored in the NYNEX Portholes system[8][9] where the video media was used to calculate an activity measure.

The issues of different types of awareness in shared workspaces and real-time distributed groupware are described in detail in e.g. [6] and [5].

5. SOLUTIONS FOR DIFFERENT MEDIA

In this section we describe each media of Marratech Pro in more detail and evaluate different possible solutions for supporting these media on a mobile phone. In the next section we describe the actual implementation of some of these ideas.

5.1 Chat

The chat media is by far the easiest to support, and it was consequently already supported by MIM before this work started.

5.2 Shared whiteboard

The shared whiteboard tool is one of the most complex parts of Marratech Pro. A user can draw text, lines and other geometrical objects, import text and Microsoft Powerpoint and Word documents, etc. Although supporting the full functionality of the tool in a mobile device would be very hard, only displaying the whiteboard pages are much easier. The pages can be rendered as images which are then transferred and viewed on the device. Depending on the memory available on the device, the initial image could either be a high resolution version of the whole page or a scaled down version.

5.3 Shared browser

When a user distributes a web page the complete page is downloaded and distributed to all Marratech Pro clients using a reliable multicast protocol and cached locally. If the client devices supports a web browser that can be controlled by the MIM client we can allow the client to connect to the Marratech Pro cache and view the page.

Even if the client would not be able to support a web browser, there are still some options left. It is possible to render the web pages

in some simpler format that the device understands before transferring them, such as images or transforming them to WML, Wireless Markup Language.

5.4 Audio

Audio often is the most important part of a e-meeting/video conference, but it is also the hardest media to support on a limited mobile device since it can require a continuous audio stream with a relatively high bandwidth and requires a device capable of decoding and playing the audio.

If there is enough bandwidth to stream the audio and the device is capable of decoding and playing it would be possible to support audio directly. Unfortunately, the codecs with highest compression ratio for a given quality is often the ones with highest CPU requirements; without hardware support most advanced codecs is impossible to use on e.g. Java phones. Also, direct streaming doesn't give any information about communication that happens while the mobile device does not have network connectivity.

If participants in the e-meeting are not continuously speaking, it is possible to record the short audio clips, convert it to a format that the target device understands and either stream it or download it to the device and play it.

A phone gateway, such as the SystemBase Dialgate-2010 or the Cisco VG248 Analog Phone Gateway products, which the mobile user can call up an on-going session can be used to provide the user with audio-only access to an e-meeting session. Unfortunately, current mobile phones do not work with GSM and GPRS at the same time, making it impossible to support audio at the same time as other tools.

Speech recognition is another technology that could potentially be used. One problem with this technology is that participants can talk in different languages which makes speech recognition much harder. For example, in our e-corridor both Swedish and English speech is common. Other information such as the tone of the voice, laughter etc. is also hard to convey. Still, speech recognition can provide much information to the user that otherwise would be unavailable. Some speech recognition systems also require training which might be cumbersome in a dynamic group setting.

For current mobile phones the last solution, speech recognition, is the only option that provides information about the content in the audio stream since the target devices does not support audio at all. Activity indicators can at least provide information about who have said something.

5.5 Video

Similar to audio, video is hard to support due to its high bandwidth and continuous nature. While all current Java phones can display images, some only have a black and white or a gray-scale display or low resolution display. The current standards do not directly support any video format.

The limited display size makes it impossible to show more than one video stream at the same time, reducing the bandwidth that is needed. Each stream can be transcoded into a format and bandwidth that the target device can handle.

In [3] the required frame rate for video conferencing is reported as being at least 5 fps. If we are only interested in providing aware-

ness, [4] states that a frame rate as low as one snapshot every five minutes can provide group awareness in a work environment. By allowing the user to choose when to update each video snapshot the necessary bandwidth can be controlled.

6. IMPLEMENTATION

The mobility support is achieved by adding a MIM plugin to Marratech Pro. This plugin collects activity information for the activity indicators, video snapshots, images from the shared whiteboard, forwards chat messages and so on. The different mobile clients connect to the MIM server, which forwards requests and information to and from the appropriate plugins. By using the Marratech Pro MIM plugin the user always appear as a single member in the session, independent from the current point of access. See figure 3 for a view of the overall architecture.

The system operates in a pull mode: clients always have to request information from the MIM server. Requests are possible for different information such as a complete update of new messages and status changes of users, a video snapshot or an activity measure of a member of an e-meeting session. The pull mode is necessary due to the limitations discussed in section 3, but it also have some benefits. In mobile networks users often have to pay for data transmitted and this cost can be quite substantial. By only sending requests as a result of user action the user have much more control over how much data is transferred and thus the cost. It also means that the client does not have to run continuously and can therefore save battery life.

6.1 Changes to MIM

For Instant Messaging systems, each contact is mapped to a MIM user. In some systems, such as IRC or an e-meeting session, one MIM user can represent a group instead, such as an IRC channel or the Marratech Pro session. In this case, the MIM user also have sub-users which represent the members of for example the IRC channel or the Marratech Pro session. These users and sub-users which represent Marratech Pro sessions and members need some special options that are not available for regular chat or instant messaging systems.

For example, it is possible to request a video snapshot for either a single member or for all members at the same time. Similarly, it is possible to request activity indicators. It is also necessary to be able to get a list of available whiteboard pages and to request a specific page.

6.2 Media

In this section we present details about how each media is handled by the system.

Chat The chat media is already supported in the MIM system, so no additional support is needed. It is a very low bandwidth, text only media which do not need any special care.

Shared whiteboard The client requests a scaled down image of the complete whiteboard page from the Marratech Pro MIM plugin in which the user can zoom in on selected parts. Scaling is always done by the plugin, which means that each zoom or scroll request means a new request to the MIM Server.



Figure 4: Examples of the shared web browser, whiteboard, user list and combined activity and video display running on a Nokia 7650 phone

Shared browser The shared web browser is implemented using the ReqwirelessWeb² web browser component for MIDP.

On devices with limited memory only the actual HTTP address is sent to the client and the user have to manually copy it to a web browser to view it.

The user can also send URLs to the Marratech Pro MIM plugin which distributes the page to the other users.

Audio No audio support is currently implemented.

Video Requests for video snapshots are forwarded to the Marratech Pro MIM plugin, which encodes the current thumbnail image and encodes it as PNG image. This image is then sent back to the requesting client. Snapshots for different users can be requested at the same time, making it possible to retrieve the whole thumbnail view in one request. The thumbnail image size is 88x72 pixels for PAL video streams.

See figure 4 for example screenshots of the different tools.

6.3 Activity indicators

The activity of a participant is calculated passively by the Marratech Pro MIM plugin using all of the available media. The activity calculation works with any clients that use the same protocols; such as H.261 video or PCM audio over RTP[12]. The chat, whiteboard and browser media is unfortunately not standardized and the activity measure will therefore only take these media in account for participants using the Marratech Pro client. The reasons for only calculating activity passively at the client are mainly two-fold: ease of deployment and standards compliance. A solution where each

²<http://www.reqwireless.com/>

client application is responsible for providing an activity measure could provide with more accurate measure, for example by detecting mouse and keyboard activity in any application on the computer.

6.3.1 Chat and shared whiteboard and browser

The chat, shared whiteboard and the shared web browser media are all similarly handled. Since these media are only sent when a user is active, any received event increases the measured activity of that user. These media use a reliable transport protocol, so the events have to be captured in the application layer and not on the network layer where repair packets can be received from the client application of an inactive user.

6.3.2 Audio

Audio is commonly only sent when the user is active as well, although sometimes users forgets to turn of their microphone and will transmit background noise even if they're not even nearby their computer. As silence suppression is on by default in Marratech Pro this was not a big problem in the prototype and any incoming audio packet could directly add to the activity measure without having to be further processed. A solution that filters out background noise and only report activity when the audio level is above a certain threshold could easily be implemented.

6.3.3 Video

Video are normally sent continuously, so the actual existence of video packets is not enough to get any measure of activity; instead the actual content of the stream have to be taken into account.

Every decoded frame of the thumbnails are compare to the earlier frame, and a weighted sum of the absolute difference is calculated for the all pixels of the decoded frames. Although this simple algorithm works fine alternatives are needed since the CPU usage jumped from around 5-10% up to 40-50% on a 2GHz Windows XP computer, in a session with 12 participants. This algorithm is similar to the one used in NYNEX Portholes[9].

The middle part of the video is given a larger weight than the rest since this is where the face of the user is normally located and thus the most important part.

There are certain cases where this algorithm does not work. For example, if someone is in their office but just not moving enough the above algorithm will not detect any activity. Another case is where the camera is directed so that the monitor or some other always changing object are visible; in this case the algorithm will report a continuous activity. One solution to both of these cases are to use either face or skin color recognition to detect the presence of a user.

6.3.4 Display of activity measure

See figure 5 for an example of an activity indicator. The display is divided into three parts: one for the last 24 hours, one for the last 60 minutes and one for the last 60 seconds. Different activity is showed in different colors. Yellow represents video activity, green audio activity, red chat activity. The top figure represents the activity for a user that joined the session about 33 minutes ago. 30 minutes ago the user sent some audio, and a few minutes later at least one chat message. In the middle figure represents an user that have the camera located far from the computer, which makes it hard for the algorithm to detect any activity as the user works in front of the computer. The high spikes shows when the user entered or left

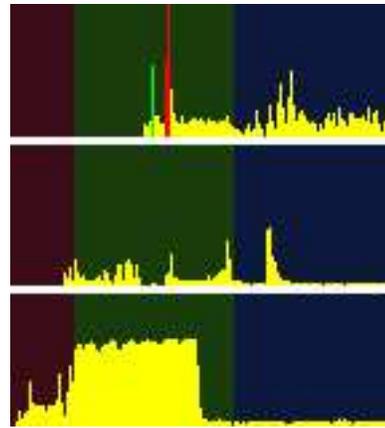


Figure 5: Examples of activity indicators

the office. The final figure shows an example where the algorithm fails. The constant activity shown is due to the camera being directed towards the computer monitor, and the sharp fall in activity happens when the screen saver turns on.

7. FUTURE WORK AND DISCUSSION

An obvious improvement would be to implement some form of audio support when devices that implement new standards such as MIDP 2.0 and MMAPi are available.

The same solutions provided in this paper would also be useful for another form of limited clients with similar problems: web browsers. For example, public web terminals at airports, web cafes or other instances where a user is not able to install or run the regular e-meeting software. WebSmile[7] and NYNEX Portholes[8] are examples of existing systems with web interfaces to video streams. Activity indicators are not only useful for users on mobile devices; participants using the regular e-meeting client would also benefit from this tool.

As shown in section 6.3 the activity indicators do show real activity most of the time, although both the actual measuring and the display can be improved. The most immediate drawback of the activity indicators are the high CPU usage of the algorithm, especially when there are many participants.

The main problem with the usability of the system is due to the slow network speeds. Downloading one video snapshot takes around 6 seconds using GPRS, making it hard to quickly get an overview of a group of people. By using JPEG format instead of PNG the typical image size decreased from around 15000 to 2000 bytes, reducing the download time for one video snapshot or a whiteboard image to under 2 seconds. Since JPEG is not supported by the MIDP standard a JPEG decoder is needed, increasing the size and complexity of the client.

8. CONCLUSIONS

In this paper we have presented the design and implementation of a novel system for supporting e-meetings on a limited device such as a mobile phone. Of the existing media in the commercial e-meeting application Marratech Pro we currently support all except for audio, which is missing due to the current limitations of the devices existing on the market today. The limitations are over-

come by using an existing system, MIM, to which we add a plugin implemented inside the Marratech Pro application that handles transcoding of media and adds HTTP support to make it suitable for a mobile phone. The final system is transparent to other users, requiring no new support from other client applications making it easy to deploy.

In addition to supporting most of the existing media, we add activity indicators to provide the user with information that is otherwise unavailable as a mean to compensate for the missing media. Activity indicators show the user the recent activity of other participants, making it possible to quickly get a sense of the current and recent status of other users.

Acknowledgements

This work was supported by the Centre for Distance-spanning Technology (CDT) under the RadioSphere and VITAL projects.

9. REFERENCES

- [1] JavaTM 2 Platform: Micro Edition, A White Paper. Sun Microsystems, Inc.
- [2] E. Amir, S. McCanne, and H. Zhang. An application level video gateway. In *Proceedings of ACM Multimedia 1997*.
- [3] M. Chen. Achieving effective floor control with a low-bandwidth gesture-sensitive videoconferencing system. In *Proceedings of ACM Multimedia 2002*.
- [4] P. Dourish and S. Bly. Portholes: supporting awareness in a distributed work group. In *Proceedings of CHI 92*.
- [5] C. Gutwin. *Workspace Awareness in Real-Time Distributed Groupware*. PhD thesis, University of Calgary, 1997.
- [6] C. Gutwin, S. Greenberg, and M. Roseman. Workspace awareness in real-time distributed groupware: Framework, widgets, and evaluation. In *People and Computers XI (Proceedings of HCI '96)*.
- [7] M. Johanson. A RTP to HTTP video gateway. In *Proceedings of the tenth international conference on World Wide Web*, 2001.
- [8] A. Lee, A. Girgensohn, and K. Schlueter. NYNEX Portholes: Initial User Reactions and Redesign Implications. In *GROUP'97, Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work*. ACM Press, 1997.
- [9] A. Lee, K. Schlueter, and A. Girgensohn. Sensing activity in video images. In *CHI 97 Extended Abstracts*. ACM Press, 1997.
- [10] P. Parnes, K. Synnes, and D. Schefström. Lightweight application level multicast tunneling using mtunnel. *Journal of Computer Communication*, 1998.
- [11] R. Parviainen and P. Parnes. Mobile instant messaging. In *Proceedings of the 10th International Conference on Telecommunications ICT'2003*.
- [12] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications, 1996. IETF RFC1889.
- [13] Mobile Information Device Profile Final Specification 1.0a. JSR-000037, Sun Microsystems, Inc.
- [14] Mobile Information Device Profile Final Specification 2.0. JSR-000118, Sun Microsystems, Inc.
- [15] Mobile Media API specification. JSR-000135, Sun Microsystems, Inc.
- [16] T. Turetli and J. Bolot. Issues with multicast video distribution in heterogeneous packet networks. In *Proceedings of the Sixth International Workshop on Packet Video*, 1994.
- [17] Wireless Java: Carrier and OEM Support Guarantees Ubiquity, 2002. Zelos Group, <URL:<http://www.zelosgroup.com/expertise/wirelessjava.htm>>.