Chapter 12

Meeting browsers and meeting assistants

Steve Whittaker, Simon Tucker, Denis Lalanne

The previous chapter (Chapter 11) explained how user requirements directed our development of meeting support technology, more specifically meeting browsers and assistants. Chapters 3 to 9 discussed the enabling components, i.e. the multimodal signal processing necessary to build meeting support technology. In the following, we will present an overview of the meeting browsers and assistants developed both in AMI and related projects, as well as outside this consortium.

12.1 Introduction

Face to face meetings are a key method by which organizations create and share knowledge, and the last 20 years have seen the development of new computational technology to support them.

Early research on meeting support technology focused on group decision support systems [Poole and DeSanctis, 1989], and on shared whiteboards and large displays to promote richer forms of collaboration [Mantei, 1988, Moran et al., 1998, Olson et al., 1992, Whittaker and Schwarz, 1995, Whittaker et al., 1999]. There were also attempts at devising methods for evaluating these systems [Olson et al., 1992]. Subsequent research was inspired by ubiquitous computing [Streitz et al., 1998, Yu et al., 2000], focusing on direct integration of collaborative computing into existing work practices and artifacts. While much of this prior work has addressed support for real time collaboration by providing richer interaction resources, another important research area is interaction capture and retrieval.

Interaction capture and retrieval is motivated by the observation that much valuable information exchanged in workplace interactions is never recorded, leading people to forget key decisions or repeat prior discussions. Its aim is to provide computational techniques for analyzing records of interactions, allowing straightforward access to prior critical information. Interaction capture is clearly a difficult problem. A great deal of technology has already been developed to support it [Brotherton et al., 1998, Mantei, 1988, Moran et al., 1997, 1998, Whittaker et al., 1994], but these systems have yet to be widely used.

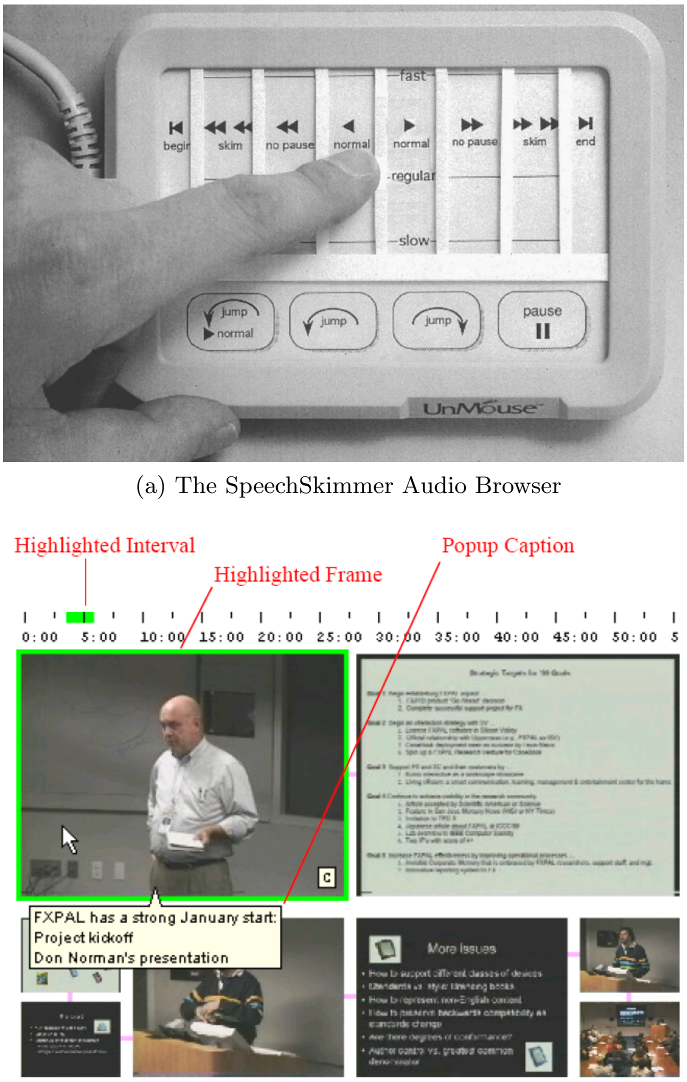
In this chapter, we will consider two main categories of meeting support technology, in relation to the requirements elicited in Chapter 11. We first describe interaction capture and retrieval systems, and then live meeting assistants that have been the focus of more recent research. The first category comprises systems that are designed to enable users to process and understand meeting content, generally after the meeting has taken place. We will present various *meeting browsers*, i.e. user interfaces that support meeting browsing and search, for instance for a person who could not attend a meeting. In contrast, *meeting assistants*, introduced afterwards, are designed to support the real-time meeting process, aiming to increase interaction quality, productivity or decision making within the meeting itself.

12.2 Meeting browsers

12.2.1 Categorization of meeting browsers

It is possible to categorize different meeting browsers – within interaction capture and retrieval systems – in terms of browser focus [Tucker and Whittaker, 2004]. Focus is defined as the main device for navigating the data, or the primary mode of presenting meeting data. We identified four main classes of meeting browsers, shown in Table 12.1. Moreover, two classes can be considered as perceptual and two others as semantic, depending on the level of analysis they require.

The first class of browsers focus on audio, including both presentation [Degen et al., 1992, Hindus and Schmandt, 1992] and navigation via audio [Arons, 1997]. Others focus on video: examples including video presentation [Girgensohm et al., 2001] or video used for navigation [Christel et al., 1998]. The third class of browsers presents meeting artifacts. Meeting artifacts may be notes made during the meeting, slides presented, whiteboard annotations Cutler et al. [2002] or documents examined in the meeting. All of these can be used for presentation and access. A final class of browser focuses on derived data such as a transcript generated by applying automatic speech recognition (ASR) to a recording of the interaction. Other derived data might include: entities extracted from the recording (names, dates or decisions), emotions, or speech acts Lalanne et al. [2003]. We call this final class discourse browsers because their focus is on the nature of the interaction.

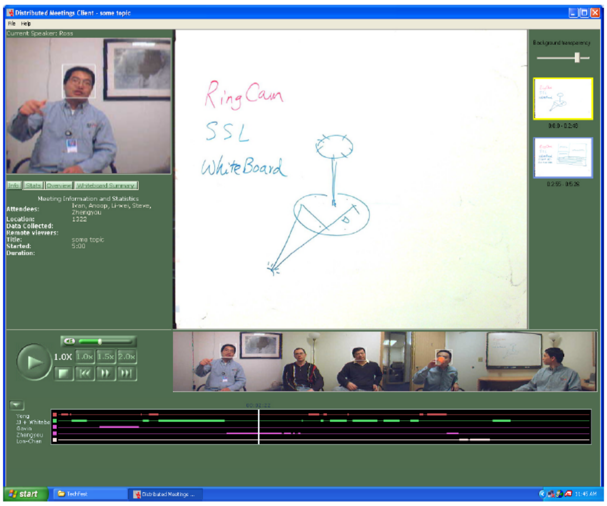


(a) The SpeechSkimmer Audio Browser

(b) The Manga Video Browser

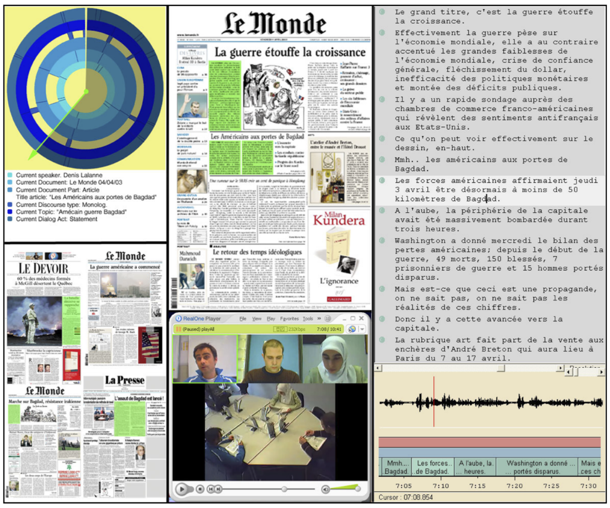
Figure 12.1: Audio and video browsers. Reprinted by permission of the publishers, respectively from Arons [1997] and **?**.

An example of an audio browser is SpeechSkimmer [Arons, 1997] shown in Figure 12.1a. Here the device allows the user to browse audio at four different levels of compression – these levels being determined by acoustic properties of the audio source. For example, at the third level only 5 seconds of speech following significant pauses is played back to the user, the significant pause being used here to define a new 'unit' of discourse. On top of this acoustic segmentation, the user can alter the playback speed and control the audio stream. This allows the user to quickly navigate to and browse relevant portions of the audio. Figure 12.1b shows an example video browser [**?**Girgensohm et al., 2001]. These browsers are typically centered around keyframes, static images which are used to represent a portion of the video. The Manga Video Browser shown in Figure 12.1b took this further and used the size of keyframes to indicate the relevance of the corresponding video portion. Thus the Manga display is similar to a comic book, drawing the user towards the interesting parts of the video.


Distributed Meetings Client - some topic
File Help
Current Speaker: Ross
Info Stats Overview Whiteboard Summary
Ring Cam
SSL
White Board
1.0X 1.0x 1.5x 2.0x
start

Meeting Information and Statistics
Attendees: Ivan, Anoop, Li-wei, Steve, Zhengyou
Location: 1322
Data Collected:
Remote viewers:
Title: some topic
Started: 5:00
Duration:

(a) An artifact browser focused on a shared whiteboard. Reprinted from Cutler et al. [2002], by permission of the publisher.



- Current speaker. Denis Lalanne
- Current Document: Le Monde 04/04/03
- Current Document Part: Article
  Title article: "Les Américains aux portes de Bagdad"
- Current Discourse type: Monolog.
- Current Topic: "Amécain guerre Bagdad"
- Current Dialog Act: Statement

- Le grand titre, c'est la guerre étouffe la croissance.
- Effectivement la guerre pèse sur l'économie mondiale, elle a au contraire accentué les grandes faiblesses de l'économie mondiale, crise de confiance générale, fléchissement du dollar, inefficacité des politiques monétaires et montée des déficits publiques.
- Il y a un rapide sondage auprès des chambres de commerce franco-américaines qui révèlent des sentiments antifrançais aux Etats-Unis.
- Ce qu'on peut voir effectivement sur le dessin, en-haut.
- Mmh.. les américains aux portes de Bagdad.
- Les forces américaines affirmaient jeudi 3 avril être désormais à moins de 50 kilomètres de Bagdad.
- A l'aube, la périphérie de la capitale avait été massivement bombardée durant trois heures.
- Washington a donné mercredi le bilan des pertes américaines; depuis le début de la guerre, 49 morts, 150 blessés, 7 prisonniers de guerre et 15 hommes portés disparus.
- Mais est-ce que ceci est une propagande, on ne sait pas, on ne sait pas les réalités de ces chiffres.
- Donc il y a cette avancée vers la capitale.
- La rubrique art fait part de la vente aux enchères d'André Breton qui aura lieu à Paris du 7 au 17 avril.

(b) FriDoc, a discourse browser which links discourse to documents [Lalanne et al., 2003].

Figure 12.2: Artifact and discourse browsers

Perceptual	Audio	SpeechSkimmer [Arons, 1997]
	Video	Video Manga [Girgensohm et al., 2001]
Semantic	Artifact	Shared Whiteboard [Cutler et al., 2002]
	Derived data	FriDoc [Lalanne et al., 2003]

Table 12.1: Main categories of meeting browsers with examples.

Cutler et al. [2002] describe a typical artifact browser, shown in Figure 12.2a. Although it includes audio and video components, the central focus of the interface is the whiteboard display. The user is able to select annotations made on the whiteboard and navigate to the corresponding point in the meeting. The artifact in question is a community artifact since it can be altered by any of the meeting participants. Figure 12.2b shows FriDoc, a discourse browser developed by Lalanne et al. [2003]. Here the focus and means of navigation is the speech and interaction that took place in the meeting. In addition, the speech is linked to the relevant documents which were discussed and the interface is time-synchronized so the user is able to use any of the components to navigate around the meeting.

We refer to audio and video indices as perceptual since they focus on low-level analysis using signal processing methods. Artifacts and derived indices are referred to as semantic since they rely on higher-level analysis of the raw data. Perceptual and semantic systems have different underlying user models. Perceptual systems assume that users will access data by browsing audio or video media selecting regions of interest using random access. In contrast, semantic systems provide higher levels of abstraction, allowing users greater control using search, or by accessing key parts of the meeting (such as decisions and actions). A more detailed taxonomy and review of interaction capture and retrieval systems is provided by Tucker and Whittaker [2004]. Given the recent rise of discourse systems that fall within the *Derived Data* class, we discuss some specific examples in detail below.

12.2.2 Meeting browsers from the AMI Consortium

The need to address the variability of user requirements, observed in the AMI Consortium and related projects (see Chapter 11), lead to the creation of JFerret, a software platform and framework for browser design. The platform offers a customizable set of plugins or building blocks which can be hierarchically combined into a meeting browser. The platform allows synchronized playback of the signals displayed by the plugins, mainly speech, video, speaker segmentation, and slides. The JFerret framework has been used to implement several browsers, including audio-based, dialogue or document-centric ones, in AMI and related projects [Lalanne et al., 2005].

A typical instantiation of the platform, often referred to as *JFerret* *browser* [Wellner et al., 2005a,b], is shown in Figure 12.3. This browser is typical of the current state of the art, offering random access to audio and video as well as access via semantic representations such as the speech transcript and artifacts such as meeting slides. Audio and video recordings can be accessed directly using player controls. Speech is transcribed, and presented in a transcript containing formatting information showing speaker identification (signaled using color coding for each speaker). The transcript depicted in the figure is human generated and therefore contains no errors, but in general the transcript will be generated using ASR. Clicking on a particular speaker contribution in the transcript begins playing the audio and video related to that contribution. The interface also shows a profile indicating overall contributions of each of the speakers, using the same color coding. This representation can be scrolled and zoomed allowing users to form an impression of overall speaker contribution levels. Finally, the system shows accompanying artifacts including presentations and whiteboard activities. Slides are temporally indexed so that selecting a specific slide accesses other data at that point in the meeting. Whiteboard events are presented as video streams and cannot therefore be used to directly index into the meeting. The JFerret browser has been evaluated by various teams to determine its utility [e.g., Whittaker et al., 2008, Section 5].

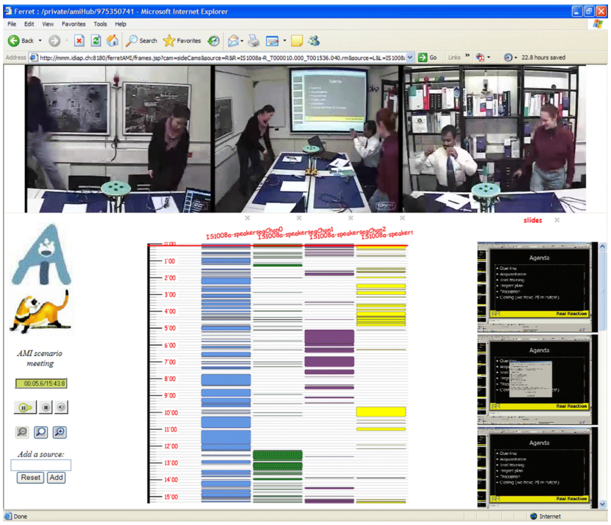


Figure 12.3: JFerret, a typical meeting browser. Reprinted by permission of the authors.

Other browsers have been implemented within the AMI and IM2 consortia, some focused on audio and speech, and others focused on more media. Three audio-based browsers [AMI, 2006] were implemented in the JFerret framework. They all provide access to audio recordings, with speaker segmentation and slides, and enhance speech browsing in two ways.

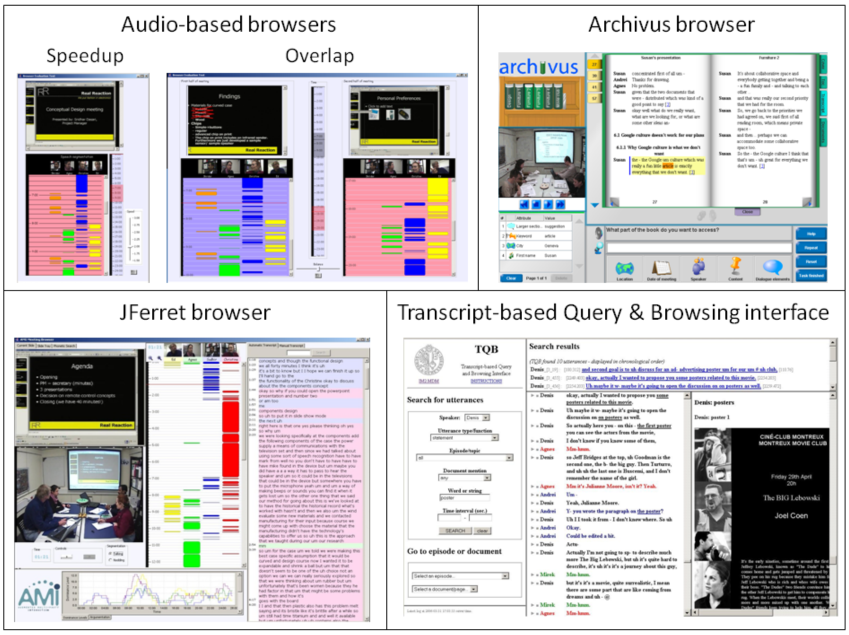


Figure 12.4: Five speech-centric meeting browsers from the AMI and IM2 consortia, illustrating the diversity of media and layouts. Components include audio, video, and slide players, along with speaker identification and segmentation, transcript, and various query parameters in Archivus and TQB. Reprinted by permission of the authors.

The *Speedup browser* accelerates audio playback while keeping speech understandable to avoid the chipmunk effect. Playback is user-controlled allowing 1.5 and 3 times normal playback rates [AMI, 2006, page 21]. The *Speedup browser* includes a timeline, scrollable speaker segmentations, a scrollable slide tray, and headshots with no live video. The speedup method has been extensively user tested and compared with other methods of speech compression, such as silence removal, unimportant word removal and unimportant phrase removal [**?**]. The *Overlap browser* achieves the compression effect in a different way by presenting two different parts of a meeting in the left vs. right audio channels, assuming that the user will take advantage of the cocktail party effect to locate the more relevant channel and then adjust the audio balance to extract the interesting facts [AMI, 2006, page 22]. Again this method was based on extensive experimentation with human subjects to validate the approach and design [Wrigley et al., 2009]. Temporal compression of speech was also used in the *Catchup browser*. Catchup allows users to join a meeting late using compression to catch up on the audio content they missed, or more generally to rapidly revisit audio content. As the previous other two, this browser was designed following careful user test-ing and shown to support comprehension of missed meeting content [Tucker et al., 2008, 2010]. Audio-based browsers require very little human preparation of automatically recorded data before use, and their performance on information extraction tasks as well as summarization is clearly encouraging (see Chapter 13 on user evaluations).

Several other browsers implemented within the AMI and IM2 consortia were focused on more media than speech. In addition to the JFerret framework and browser mentioned above, the *Transcript-based Query and Browsing (TQB) interface* [Popescu-Belis and Georgescul, 2006, Popescu-Belis et al., 2008a] is another speech-centric browser, which provides a number of manual (reference) annotations in order to test their utility for meeting browsing: manual transcript, dialogue acts, topic labels, and references to documents. These parameters can be used to formulate queries to a database of meeting recordings, and have been tested with human subjects on a fact-finding and verification task (see Chapter 13). The conclusions are also used to set priorities for research on the automatic annotation of these parameters on meeting data.

Archivus [Ailomaa et al., 2006, Melichar, 2008] is a partially implemented meeting browser that supports multimodal human-computer dialogue. Its purpose was to gather user requirements [Lisowska et al., 2007] especially with respect to modality choice, using a Wizard-of-Oz approach. Archivus uses reference transcripts enriched with annotations (speaker segmentation, topic labels, documents) to answer user queries that are expressed as a set of attribute/value constraints over one or several meetings. An implementation using a standalone dialogue engine with a multilingual front-end and a touch-screen on a mobile device was built for a subset of the Archivus search attributes, as the *Multilingual Multimodal Meeting Calendar (M3C)* [Tsourakis et al., 2008].

FriDoc [**?**] and *JFriDoc* [Rigamonti et al., 2006], are document-centric browsers that link documents discussed during the meeting, dialogue transcripts, slides and audio-video streams. They exploit automatic alignments between printed documents and speech as well as video (see Figure 12.5), highlighting when a document section was discussed during a meeting (by automatic alignment of document content with speech transcript content), or when a document was the visual focus (by automatic alignment of document image with video of projection screen, or document on the table). In these browsers, clicking on a specific document part (e.g. a section, an image, etc.) accesses the audio/video recording at the moment when the content of that document section is being discussed. In the same way selecting a moment in the audio/video stream will automatically select the relevant document section. The benefit of this automatic alignment has been evaluated, and proven to be useful for meeting browsing, using the methods described in Chapter 13.

Similarly, *ViCoDe* (*Video Content Description and Exploration*) com-

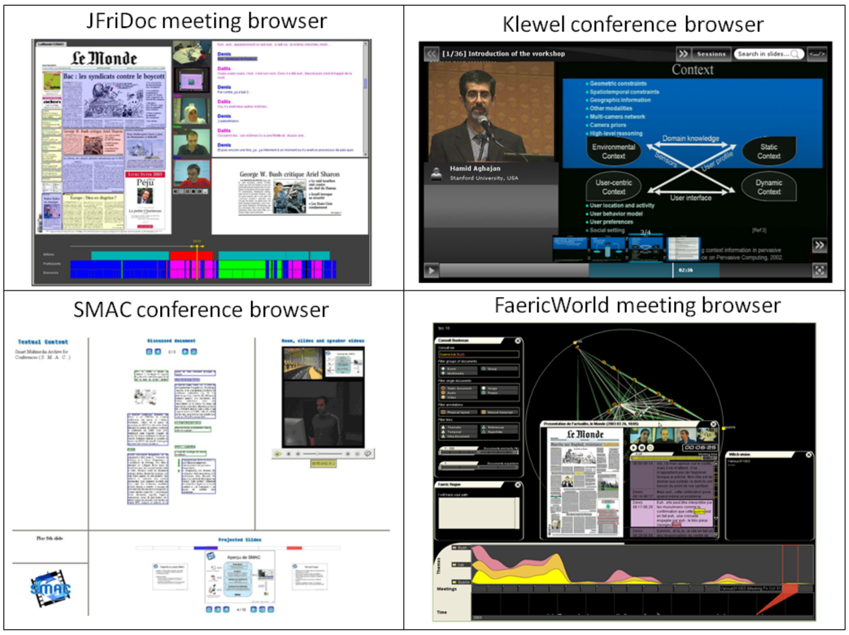


Figure 12.5: Document-centric meeting browsers and conference browsers from the AMI and IM2 consortia described in the text. Document/speech alignment is central to all layouts. Reprinted by permission of the authors.

putes the similarity of speech and document sentences. When combined with relevance feedback this supports new ways of browsing meetings [Marchand-Maillet and Bruno, 2005]. *FaericWorld* [Rigamonti et al., 2007] enhances document-based browsing with cross-meeting representations of documents and links. For each collection of meetings, links between all multimedia data associated with the meetings are automatically derived through an analysis of the input streams upon indexing of the meeting into the system's database. Users can then query the system with full text search or directly browse through links, using interactive visualizations. Finally, *WotanEye* [Évequoz and Lalanne, 2009] enables ego-centric access to meeting fragments using personal cues, such as the user's social network.

An extension of the discourse browsing approach includes the analysis and presentation of an entire whole meeting through some form of summarization, for instance as presented in Chapter 10. Variants on this include analyzing the meeting to identify important discourse acts, allowing users to focus directly on decisions or on items to do [Fernández et al., 2008]. Another approach has been exemplified by the Summary Visualizer (SuVi), which uses the automatic extractive or abstractive summaries based on ASR, together with video information, to create a multimodal storyboard (or comic book) meeting summary [?]. The output can be visualized and printed, but can also be used in HTML format within a more complex meeting browser.

12.2.3 Conference recording and browsing

Despite the large number of research prototypes, there are still no commercially available end-user meeting browsers. This is all the more surprising since some of the commercially available systems for co-ordinating remote meetings offer recording capabilities, but no support for more advanced browsing (other than replay). The meeting browsers developed within AMI and related projects have evolved towards two *end-user products*, but for a slightly different task, namely conference recording and browsing. The two systems answer a growing need for conference recording in flexible settings and playback using cross-platform, user-friendly interfaces, as initiated for instance in the Classroom 2000 educational environment [Abowd, 1999]. These two applications to conference recording and browsing use fewer capture devices than instrumented meeting rooms, and off-the-shelf technology rather than capture devices designed on purpose, resulting in smaller amounts of data to store and process, which might explain why they were quicker to reach product stage.

One system is commercialized through spin-off company of the Idiap Research Institute named Klewel (http://www.klewel.com),[1] while the other one was developed by the University of Fribourg and the CERN in Geneva within the SMAC project (Smart Multimedia Archive for Conferences, http://smac.hefr.ch) and is in use at these institutions. Both systems extract a number of robust indexes, such as slide changes, text from slides, and slide/audio/video synchronization, which are helpful for browsing, and provide some support for fact-finding. The SMAC system, in addition, is able to automatically hyperlink the fragments of the scientific article that is being presented to the related audio-video sequence [Lalanne et al., 2004]. Such technologies derived from our consortia research give these browsers an advantage over other competing systems [Herr et al., 2010].

12.3 Meeting assistants: real-time meeting support

To demonstrate how component technologies might be combined to address some of the user requirements presented in the previous chapter (see Chapter 11), several other applications have been designed and implemented by members of the AMI Consortium. Although our initial focus on meeting

[1] The Klewel/Idiap presentation acquisition system has been adopted by the ACM Digital Media Capture Committee, following the successful recording and distribution of the CHI 2007 conference. The company has received the European Seal of e-Excellence, from the European Multimedia Forum, at CeBIT 2008.

browsers, we shifted toward real-time meeting assistants that aim to increase the efficiency of an ongoing meeting. The achievements thus cover the multiple facets of meeting support addressing user needs before, during, and after a meeting (see Chapter 11).

Several pieces of software infrastructure were designed to support the implementation of demonstrators, among which are the three quoted in this section. The Hub is a subscription-based client/server mechanism for real-time annotation exchange [AMIDA, 2007]. The Hub allows the connection of heterogeneous software modules, which may operate remotely, ensuring that data exchange is extremely fast – a requirement for real-time meeting support. Data circulating through the Hub is formatted as timed triples (time, object, attribute, value), and is also stored in a special-purpose database, which was designed to deal with large-scale, real-time annotations and metadata of audio and video recordings. 'Producers' of annotations send triples to the Hub, which are received by the 'consumers' that subscribe to the respective types; consumers can also query the Hub for past annotations and metadata about meetings. The HMI Media Server [see **?**] complements the Hub for media exchange. It can broadcast audio and video captured in an instrumented meeting room to various 'consumers', thus allowing a more flexible design of interfaces that combine the rendering of media streams with annotations and metadata. The server is built on low-level DirectShow filters under Microsoft Windows, thus providing accessible interfaces in C++ and Java, and can stream media over UDP network ports to multiple targets.

12.3.1 Improving user engagement in meetings

An important requirement for meeting assistants is to improve the meeting experience for participants attending remotely. The objective is to go beyond simply exchanging audio and video between remote participant(s) and physically co-located ones. We use AMI processing technologies to enrich the audio and video with information to help remote participant(s) to better understand the communication going on within the meeting, allowing them to intervene more efficiently in the discussion. Two such meeting support applications were designed by AMI Consortium members: one intended for users connected through a mobile device, and the other for users connected through a desktop or laptop computer.

The Mobile Meeting Assistant (MMA) is a prototype mobile interface aimed at improving remote access to meetings [Matena et al., 2008]. Remote participants often complain that they have little idea about the underlying interpersonal dynamics of meetings (e.g. gestures or eye gaze), and providing high quality video data is still not possible with today's mobile devices. Unlike more traditional teleconferencing devices, the MMA allows remote users not only to hear other participants and to view projected material (slides), but also to gain insights into their non-verbal communication. Two

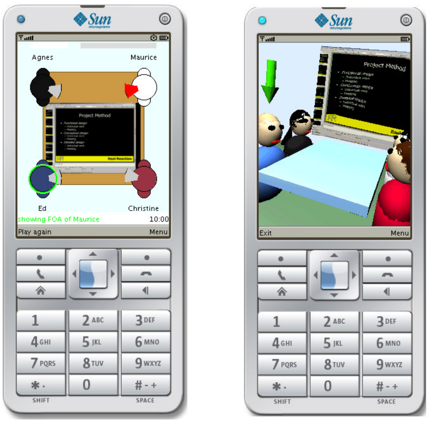
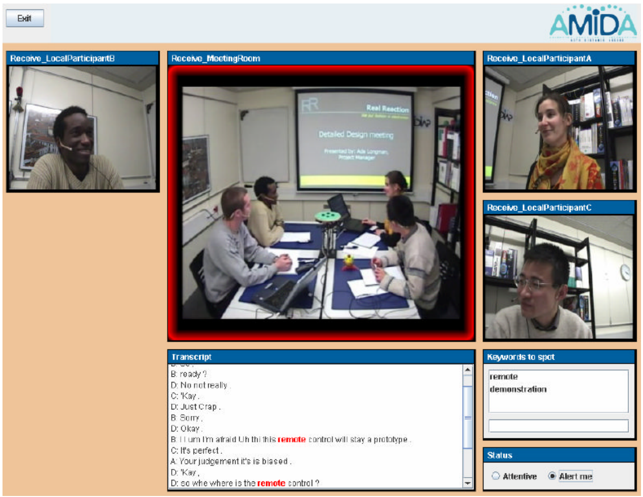


Figure 12.6: The 2D and 3D interfaces of the Mobile Meeting Assistant [Matena et al., 2008]. Reprinted by permission of the authors.

main modes were designed to display a representation of the physically-collocated group on the remote participant's mobile device: a two-dimension (2D) and a three-dimension (3D) representation, both shown in Figure 12.6.

The MMA prototype uses graphical elements to represent non-verbal information related to the audio-visual behaviors of the co-located participants, including: (1) speaking status, inferred from ASR and speaker segmentation (see Chapters 4 and 5), shown by red moving lips; (2) head orientation obtained through video processing (see Chapter 6); and (3) individual or joint visual focus of attention obtained through multimodal processing (see Chapter 6 and Section 9.3.1), represented in the 3D view by a green arrow. A user evaluation was performed using a meeting from the AMI Corpus (see Chapter 2) with a small group of subjects (13 people) who acted as remote participants [see for details Matena et al., 2008, AMIDA, 2008]. Feedback from these subjects, as well as from industrial partners in the AMI Community of Interest, suggests that the graphical conventions might be made clearer and more realistic, and that the MMA could provide richer information about the participants.

The User Engagement and Floor Control (UEFC) prototype trades mobility for higher computing power, bandwidth, and size of display [**?**]. The UEFC is motivated by the fact that, in meetings, remote participants are often multi-tasking (e.g. reading email while listening to the ongoing meeting conversation), and might benefit from receiving alerts when specific keywords are uttered, or when they are addressed by one of the co-located group's members. The UEFC integrates keyword spotting to support alerts for selected keywords, visual focus of attention, and online addressee detection, which provides alerts about when the remote participant's image becomes the focus of attention of local participants – the interface of the


Exit
AMIDA
Receive_LocalParticipantB
Receive_MeetingRoom
Receive_LocalParticipantA
Receive_LocalParticipantC
Real Reaction
Detailed Design meeting
Presented by: Ade Longman, Project Manager

Transcript

B: ready ?
D: No not really .
C: 'Kay .
D: Just Crap .
B: Sorry .
D: Okay .
B: I I um I'm afraid Uh thi this **remote** control will stay a prototype .
C: It's perfect .
A: Your judgement it's is biased .
D: 'Kay .
D: so whe where is the **remote** control ?

Keywords to spot

remote
demonstration

Status

○ Attentive ● Alert me

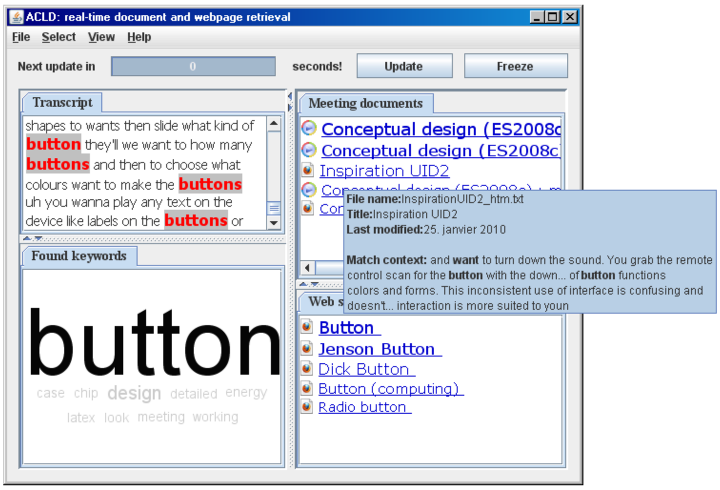
Figure 12.7: The User Engagement and Floor Control System [?]. Reprinted by permission of the authors.

UEFC system is shown in Figure 12.7. A dedicated addressee detector uses lexical features from the ASR, and the output of the visual focus of attention analyzer, for a binary decision task (whether the remote participant is being addressed or not). The online dialogue act segmentation and labeling (see Chapter 8) are also integrated.

12.3.2 Suggesting relevant documents during meetings

Participants in meetings often need access to project reference materials (e.g. meeting minutes, presentations, contracts, specification documents) but they usually do not have the time during the meeting to search for these. Similarly, they may want access to recordings of their past meetings, but again do not want to disrupt a meeting to access these. The Automatic Content Linking Device [ACLD, see Popescu-Belis et al., 2008b, ?] is a meeting support application that provides just-in-time and query-free access [as in ?Rhodes and Maes, 2000] to potentially relevant documents or recorded meetings. The ACLD thus provides automatic real-time access to a group's history, as suggestions made during an ongoing meeting.

The ACLD makes use of speech-oriented AMI core technologies such as automatic speech recognition and keyword spotting (see Chapter 5) in an instrumented meeting room and speaker diarization (Chapter 4), using the Hub to exchange annotation and the HMI Media Server to broadcast media. The main ACLD component is the Query Aggregator, which performs document searches at regular time intervals over a database of previous documents and meeting transcripts (e.g. from the AMI Corpus described in Chapter 2), using words and terms that are recognized automatically from



shapes to wants then slide what kind of **button** they'll we want to how many **buttons** and then to choose what colours want to make the **buttons** uh you wanna play any text on the device like labels on the **buttons** or

File name: InspirationUID2_htm.txt
Title: Inspiration UID2
Last modified:25. janvier 2010

Match context: and **want** to turn down the sound. You grab the remote control scan for the **button** with the down... of **button** functions colors and forms. This inconsistent use of interface is confusing and doesn't... interaction is more suited to youn

Figure 12.8: User interface of the Automatic Content Linking Device [?]. Reprinted by permission of the authors.

the meeting discussion. While the first prototypes used Apache Lucene for keyword-based search in local repositories, a more recent version use "semantic search" to cope with noise in ASR and improve the relevance of search results [?]. The Query Aggregator is also connected to the Google search engine, and separately manages a list of the top hits retrieved from a user-specified web domain using queries again based on ASR words.

The ACLD result shown to users is a list of document names ordered by relevance, refreshed at regular intervals (15 seconds) or on demand, based on the search results and on a persistence model which ensures that documents that are often retrieved persist at the top of the list. The snapshot in Figure 12.8 shows the user interface of the ACLD in a detailed view, with all four widgets visible: ASR words, tag cloud of keywords, document results (with pop-up window open when hovering over a name), and Web results. An unobtrusive view can display the widgets as superposed tabs, freeing up screen real-estate for other activities. Evaluation results for the ACLD have shown that users clicked on a suggested document every 5-10 minutes, that they found the UI "acceptably" usable, and that results of semantic search are found more relevant than those of keyword-based search five times more often.

12.4 Summary and further reading

This chapter presented two types of meeting support technologies answering some of the most important requirements that were found by the AMI Consortium and other projects (see Chapter 11). The first type (meeting browsers) supports capture, post hoc analysis and replay of meetings, whereas the second one (meeting assistants) is used during meetings to enrich live interactions between meeting participants. Several meeting browsers have been described, making use of raw video and audio recordings, of artifacts such as whiteboard recordings or documents projected or discussed during the meeting, or using annotations derived from raw data recordings, such as the speech transcript or the visual focus of attention.

Despite the number of research prototypes for meeting browsing, none of them have achieved large scale mass adoption. One reason for this lack of uptake is socio-technical issues that have to be addressed before systems become acceptable. For instance, in various user studies [e.g. starting with Whittaker et al., 1994], users expressed concerns about privacy, and about the impact of being recorded on the process of the meeting itself. This is possibly one of the reasons why, from the numerous browsers developed by AMI and related projects, the two resulting end-user products are those aimed at the recording and browsing of public conferences.

There are a number of important practical and research issues arising. For meeting browsers the technology is relatively well understood, but two main areas remain to be addressed. The first concerns data capture: basic approaches to recording high quality multimedia data are not standardized, with most meeting rooms currently lacking recording equipment. Without such data we cannot build successful browsers. The second issue relates to user value: meeting participants seem remarkably resistant to changing meeting practices, and in many studies have not embraced the opportunity to re-access recordings of past meetings[Whittaker et al., 2008]. We need a better understanding of why this is the case, as well as what situations and contexts in which participants would value such access.

Turning to real-time assistants, here the field is much more open to developing new types of tools based on analyses of ongoing behaviour. Such analyses might extend to complex dialogue issues such as conflict and debate, that might improve fundamental meeting processes. New systems might identify if particular participants are dominating a discussion or whether a discussion is leading to an unresolvable impasse. They might detect when there are implicit disagreements or help participants better understand their common ground. Again however the history of prior work has shown that meeting interactions are highly sensitive to disruption so any new technology must be designed to integrate well with existing meeting practices.

Bibliography

Gregory D. Abowd. Classroom 2000: An experiment with the instrumentation of a living educational environment. *IBM Systems Journal*, 38(4): 508–530, 1999.

Marita Ailomaa, Miroslav Melichar, Martin Rajman, Agnes Lisowska, and Susan Armstrong. Archivus: a multimodal system for multimedia meeting browsing and retrieval. In *COLING/ACL 2006 Interactive Presentation Sessions*, pages 49–52, Sydney, 2006.

AMI. Meeting browser evaluation report. Deliverable D6.4, AMI Integrated Project FP6 506811 (Augmented Multi-party Interaction), December 2006.

AMIDA. Commercial component definition. Deliverable D6.6 (ex D7.2), AMIDA Integrated Project FP7 IST033812 (Augmented Multi-party Interaction with Distance Access), `http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/` November 2007.

AMIDA. AMIDA proof-of-concept system architecture. Deliverable D6.7, AMIDA Integrated Project FP7 IST033812 (Augmented Multi-party Interaction with Distance Access), `http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/` March 2008.

B. Arons. Speechskimmer: A system for interactively skimming recorded speech. *ACM Transcations on Computer-Human Interaction*, 4(1):3–38, March 1997.

J. A. Brotherton, J. R. Bhalodia, and G. D. Abowd. Automated capture, integration and visualization of multiple media streams. In *The IEEE International Conference on Multimedia Computing And Systems*, pages 54–63, 1998.

M.G. Christel, M.A. Smith, C. Roy Taylor, and D.B. Winkler. Evolving video skims into useful multimedia abstractions. In *CHI '98*, April 1998.

R. Cutler, Y. Rui, A. Gupta, J.J. Cadiz, I. Tashev, L. He, A. Colburn, Z. Zhang, Z. Liu, and S. Silverberg. Distributed meetings: A meeting capture and broadcasting system. In *10th ACM International Conference on Multimedia*, pages 503–512, December 2002.

L. Degen, R. Mander, and G. Salomon. Working with audio: Integrating personal tape recorders and desktop computers. In *CHI '92*, pages 413–418, May 1992.

Florian Évequoz and Denis Lalanne. "I thought you would show me how to do it" – studying and supporting PIM strategy changes. In *ASIS&T 2009 Personal Information Management Workshop*, Vancouver, BC, 2009.

Raquel Fernández, Matthew Frampton, Patrick Ehlen, Matthew Purver, and Stanley Peters. Modelling and detecting decisions in multi-party dialogue. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, SIGdial '08, pages 156–163, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. ISBN 978-1-932432-17-6. URL `http://portal.acm.org/citation.cfm?id=1622064.1622095`.

A. Girgensohm, J. Borczky, and L. Wilcox. Keyframe-based user interfaces for digital video. *IEEE Computer*, 34(9):61–67, September 2001.

Monika Henziker, Bay-Wei Chang, Brian Milch, and Sergey Brin. Query-free news search. *World Wide Web: Internet and Web Information Systems*, 8:101–126, 2005.

Jeremy Herr, Robert Lougheed, and Homer A. Neal. Lecture archiving on a larger scale at the University of Michigan and CERN. *Journal of Physics: Conference Series*, 219:082003, 2010.

D. Hindus and C. Schmandt. Ubiquitous audio: Capturing spontaneous collaboration. In *1992 ACM Conference on Computer-Supported Cooperative Work*, pages 210–217, November 1992.

D. Lalanne, S. Sire, R. Ingold, A. Behera, D. Mekhaldi, and D. Rotz. A research agenda for assessing the utility of document annotations in multimedia databases of meeting recordings. In *3rd International Workshop on Multimedia Data And Document Engineering*, September 8th 2003.

Denis Lalanne and Rolf Ingold. Documents statiques et multimodalité : l'alignement temporel pour structurer des archives multimédias de réunions. *Document numrique*, 8(4):65–89, 2004.

Denis Lalanne, Dalila Mekhaldi, and Rolf Ingold. Talking about documents: revealing a missing link to multimedia meeting archives. In *Document Recognition and Retrieval XI, IS&T/SPIE's Annual Symposium on Electronic Imaging*, pages 82–91, San Jose, CA, 2004.

Denis Lalanne, Agnes Lisowska, Eric Bruno, Mike Flynn, Maria Georgescul, Mal Guillemot, Bruno Janvier, Stphane Marchand-Maillet, Mirek Melichar, Nicolas Moenne-Loccoz, Andrei Popescu-Belis, Martin Rajman, Maurizio Rigamonti, Didier von Rotz, and Pierre Wellner. The IM2 multimodal meeting browser family. Technical report, IM2 Swiss National Center of Competence in Research (Interactive Multimodal Information Management), March 2005 2005.

Agnes Lisowska, Mireille Bétrancourt, Susan Armstrong, and Martin Rajman. Minimizing modality bias when exploring input preference for multimodal systems in new domains: the Archivus case study. In *Proceedings of CHI 2007 (ACM SIGHI Conference on Human Factors in Computing Systems)*, pages 1805–1810, San Jos, CA, 2007.

M. Mantei. Capturing the capture lab concepts: A case study in the design of compute supported meeting environments. In *Conference on Computer Supported Cooperative Work*, Portland, OR, September 1988.

Stphane Marchand-Maillet and Eric Bruno. Collection guiding: A new framework for handling large multimedia collections. In *AVIVDiLib 2005 (First Workshop on Audio-visual Content And Information Visualization In Digital Libraries)*, Cortona, Italy, 2005.

Lukas Matena, Alejandro Jaimes, and Andrei Popescu-Belis. Graphical representation of meetings on mobile devices. In *MobileHCI 2008 Demonstrations (10th International Conference on Human-Computer Interaction with Mobile Devices and Services)*, Amsterdam, 2008.

Miroslav Melichar. *Design of Multimodal Dialogue-based Systems.* PhD thesis, 4081, EPF Lausanne, School of Computer and Communication Sciences, 2008. URL `http://library.epfl.ch/theses/?nr=4081`.

T.P. Moran, L. Palen, S. Harrison, P. Chiu, D. Kimber, S. Minneman, W. Melle, and P. Zellweger. "i'll get that off the audio": A case study of salvaging multimedia meeting records. In *CHI '97*, 22-27 March 1997.

T.P. Moran, W. VanMelle, and P. Chiu. Spatial interpretation of domain objects integrated into a freeform electronic whiteboard. In *Proceedings of UIST*, 1998.

Anton Nijholt, Rutger Rienks, Job Zwiers, and Dennis Reidsma. Online and off-line visualization of meeting information and meeting support. *The Visual Computer*, 22(12):965–976, 2006.

J.S. Olson, G.M. Olson, M. Storrø sten, and M. Carter. How a group-editor changes the character of a design meeting as well as its outcome. In *Proceedings of CSCW*, pages 91–92, 1992.

M.S. Poole and G. DeSanctis. Use of group decision support syatems as an appropriation process. In *HICSS Conference*, 1989.

Andrei Popescu-Belis and Maria Georgescul. TQB: Accessing multimodal data using a transcript-based query and browsing interface. In *Proceedings of LREC 2006 (5th International Conference on Language Resources and Evaluation)*, pages 1560–1565, Genova, 2006.

Andrei Popescu-Belis, Philippe Baudrion, Mike Flynn, and Pierre Wellner. Towards an objective test for meeting browsers: the BET4TQB pilot experiment. In *Proceedings of MLMI 2007 (4th Workshop on Machine Learning for Multimodal Interaction)*, LNCS 4892, pages 108–119, Brno, 2008a.

Andrei Popescu-Belis, Erik Boertjes, Jonathan Kilgour, Peter Poller, Sandro Castronovo, Theresa Wilson, Alejandro Jaimes, and Jean Carletta. The AMIDA Automatic Content Linking Device: Just-in-time document retrieval in meetings. In Andrei Popescu-Belis and Rainer Stiefelhagen, editors, *Machine Learning for Multimodal Interaction V (Proceedings of MLMI 2008, Utrecht, 8-10 September 2008)*, LNCS 5237, pages 273–284. Springer-Verlag, Berlin/Heidelberg, 2008b.

Bradley J. Rhodes and Pattie Maes. Just-in-time information retrieval agents. *IBM Systems Journal*, 39(3-4):685–704, 2000.

Maurizio Rigamonti, Denis Lalanne, Florian Évéquoz, and Rolf Ingold. Browsing multimedia archives through intra- and multimodal cross-documents links. In *Proceedings of MLMI 2005 (2nd Workshop on Machine Learning for Multimodal Interaction)*, LNCS 3869, pages 114–125. Edinburgh, UK, 2006.

Maurizio Rigamonti, Denis Lalanne, and Rolf Ingold. FaericWorld: Browsing multimedia events through static documents and links. In *Proceedings of Interact 2007 (11th IFIP TC13 International Conference on Human-Computer Interaction), Part I*, LNCS 4662, pages 102–115, Rio de Janeiro, 2007.

J. Streitz, N.A. Geisler, and T. Holmer. Roomware for cooperative buildings: Integrated design of archtectural spaces and information spaces, cooperative buildings: integrating information, organization and architecture. In *Lecture Notes In Computer Science*, volume 1370, pages 4–21. Springer, Heidelberg, 1998.

Nikos Tsourakis, Agnes Lisowska, Pierrette Bouillon, and Manny Rayner. From desktop to mobile: Adapting a successful voice interaction platform for use in mobile devices. In *SiMPE 2008 (3rd ACM MobileHCI Workshop on Speech in Mobile and Pervasive Environments)*, Amsterdam, 2008.

S. Tucker and S. Whittaker. Accessing multimodal meeting data: Systems, problems and possibilities. In *Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, June 2004.

Simon Tucker, Nicos Kyprianou, and Steve Whittaker. Time-compressing speech: Asr transcripts are an effective way to support gist extraction. In Andrei Popescu-Belis and Rainer Stiefelhagen, editors, *Machine Learning for Multimodal Interaction V (Proceedings of MLMI 2008, Utrecht, 8-10 September 2008)*, LNCS 5237, pages 226–236. Springer-Verlag, Berlin/Heidelberg, 2008.

Simon Tucker, Ofer Bergman, Anand Ramamoorthy, and Steve Whittaker. Catchup: a useful application of time-travel in meetings. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, CSCW '10, pages 99–102, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-795-0. doi: http://doi.acm.org/10.1145/1718918.1718937. URL `http://doi.acm.org/10.1145/1718918.1718937`.

Pierre Wellner, Mike Flynn, and Mal Guillemot. Browsing recorded meetings with Ferret. In Samy Bengio and Herv Bourlard, editors, *Machine Learning for Multimodal Interaction*, LNCS 3361, pages 12–21. Springer-Verlag, Berlin/Heidelberg, 2005a.

Pierre Wellner, Mike Flynn, Simon Tucker, and Steve Whittaker. A meeting browser evaluation test. In *Proceedings of CHI 2005 (ACM SIGCHI Conference on Human Factors in Computing Systems)*, pages 2021–2024, Portland, OR, 2005b.

S. Whittaker and H. Schwarz. Back to the future: pen and paper technology supports complex group coordination. In *Proceedings of CHI*, 1995.

S. Whittaker, D. Frohlich, and O. Daly-Jones. Informal communication: What is it like and how might we support it? In *Proceedings of CHI*, 1994.

S. Whittaker, J. Hirschberg, J. Choi, D. Hindle, F. Pereira, and A. Singhal. SCAN: designing and evaluating user interfaces to support retrieval from speech archives. In *Proceedings of SIGIR99 Conference on Research and Development in Information Retrieval*, pages 26–33, 1999.

Steve Whittaker, Simon Tucker, Kumutha Swampillai, and Rachel Laban. Design and evaluation of systems to support interaction capture and retrieval. *Personal and Ubiquitous Computing*, 12(3):197–221, 2008.

Stuart N. Wrigley, Simon Tucker, Guy J. Brown, and Steve Whittaker. Audio spatialisation strategies for multitasking during teleconferences. In *INTERSPEECH*, pages 2935–2938, 2009.

H. Yu, T. Tomokiyo, H. Wang, and A. Waibel. New developments in automatic meeting transcription. In *Proceedings of ICSLP*, 2000.