Chapter 11

User requirements for meeting support technology

Denis Lalanne, Andrei Popescu-Belis

There are many kinds of information technology that can be used to make meetings more productive, some of which have more to do with what happens before and after meetings than during them. Document repositories, presentation software, and even intelligent lighting can all play their part. In our discussion of user requirements, we will restrict ourselves to considering systems that draw on the multimodal signal processing techniques described in the earlier chapters of this book to capture and analyze meetings. Such a system might help people understand something about a past meeting that has been stored in an archive, but it might equally well aid meeting participants in some way during the meeting itself. It might help system users understand what has been said at a meeting, or it could serve some other purpose — conveying an idea of who was present, who spoke and what the interaction was like, for instance. We will refer to all such systems, regardless of their purpose or when they are used, as "meeting support technology".

In the classic approach to software development, user requirements are set early the process and do not change as development proceeds. This approach causes difficulties for meeting support technology, as it does for many other kinds of software, because multimodal signal processing raises the possibility of very new kinds of systems that many potential users have trouble imagining. We begin by describing the most well-known models for the software development process and explaining why they break when user requirements are hard to to define. We then introduce the approach that evolved to fit the needs of this community. In a nutshell, what is required is an iterative process that through interaction between developers and potential users, gradually narrows and refines sets of user requirements for individual applications. Finally, we both illustrate the approach and lay out specific user requirements by discussing the major user studies that have been conducted for meeting support technology.

11.1 Models for the software development process

The simplest way to think of the software development process is as a series of stages where each is complete before the next one starts. This approach is commonly known as the "waterfall model". Although different versions of the model include different stages, they always begin with requirements specification. Afterwards come design, implementation (that is, actually writing software code), testing, deployment, and maintenance of the finished product. Software is evaluated in terms of how well it meets the initial specifications ISO/IEC (2001). The argument used by proponents of the model is that following it tends to lead to higher quality software — in particular, because changing the user requirements after coding has started can waste programming time and lead to a product with an incoherent design or inconsistent and difficult to maintain implementation. In practice, however, life is often not that simple, and the software development process iterates by moving back to "prior" stages whenever the need for that becomes apparent. The most classic way of thinking of this iteration is called the "spiral model" (Boehm, 1986). In this model, the aim is still to establish stable user requirements early on, but development occurs in cycles. For each cycle, development involves elaborating the objectives of the cycle, identifying and resolving major sources of risk, defining and elaborating the product, and finally, planning the next cycle. This approach acknowledges the difficulty of following the waterfall stages and mitigates the risk to the quality of the final product through its emphasis on risk analysis.

There are many variants on the iterative design process that have been expressed as a result of grappling with how to produce different kinds of software. For meeting support technology, the main difficulty for software development stems from the fact that multimodal signal processing is so new it promises software that is a radical departure from the user's existing experience. In cases like this, there are two main tendencies that drive development. On the one hand, there is "technology push": as new technologies arise from research and reach maturity, their proponents naturally think of deploying them in products. On the other hand, "market pull", or the attempt to satisfy unmet needs that potential users have already expressed, is just as natural — and often more immediately rewarding in business terms. The problem is how to make these two tendencies meet in the middle, when potential users are not aware of what is technologically feasible, and system developers are not aware of what the users require. This problem calls for a different kind of iteration. In "user-centered design", each cycle of the iteration involves system development and evaluation, so that the software development process gradually homes in on what is possible that the users need (Dix et al., 1993). Each cycle includes requirements elicitation, software design and implementation, and finally, task-based evaluation (Sommerville, 2007, Chapter 2) in a back-and-forth exchange between users and developers.

11.2 A model for meeting support technologies

We propose the *helix shaped model* shown in Figure 11.1 as an explicit representation of the iterations that have shaped several projects in meeting support technologies, including several achievements of the AMI Consortium. The helix rotates through four sectors that form the horizontal axis, while making progress towards specific software products on the vertical axis. Although this axis is correlated with time, it is not necessary that all components of a system progress at the same pace, therefore the axis is better characterized in terms of specificity and product completeness. The four sectors of each iteration correspond to the principal stages in the development of software prototypes, especially of research ones: requirements, analysis and design, implementation, and testing. These stages match the ISO 9126 recommendations (ISO/IEC, 2001) and the IBM Rational Unified Process (Kroll and Kruchten, 2003).

Based on experience from developing meeting browsers, exemplified in the following chapter of the book, the four sectors of the helix are divided by two axes: the 'people' axis from users to developers, and the 'systems' axis from construction to evaluation. Hence, the four sectors of the helix are: requirements elicitation, design/implementation, performance evaluation (i.e. intrinsic evaluation of components by developers), and task-based evaluation (i.e. extrinsic evaluation of a product by users). The evaluation of complex systems such as meeting browsers, which rely on advanced multimodal signal processing to extract features from meeting recordings, is not so much a matter of testing/verification, but rather of measuring the users' acceptability of the actual error rate of the processing (which is inevitably non-zero for human interaction analysis).

As in other iterative software development models, the evaluation results obtained in one iteration can also be viewed as more or less specific elicited requirements, and are therefore used to derive new specifications for design in the subsequent iteration or loop. Each loop of the helix ends with a certain form of evaluation, which can be evaluation *per se*, or some kind of analysis of the product of the iteration, based on users' experience with a more or less fully implemented prototype. Depending on each iteration or loop, the evaluation methods that are used may vary considerably to match the technologies under evaluation. For instance, browser prototypes may use hand-crafted annotations or a Wizard-of-Oz, while systems that integrate

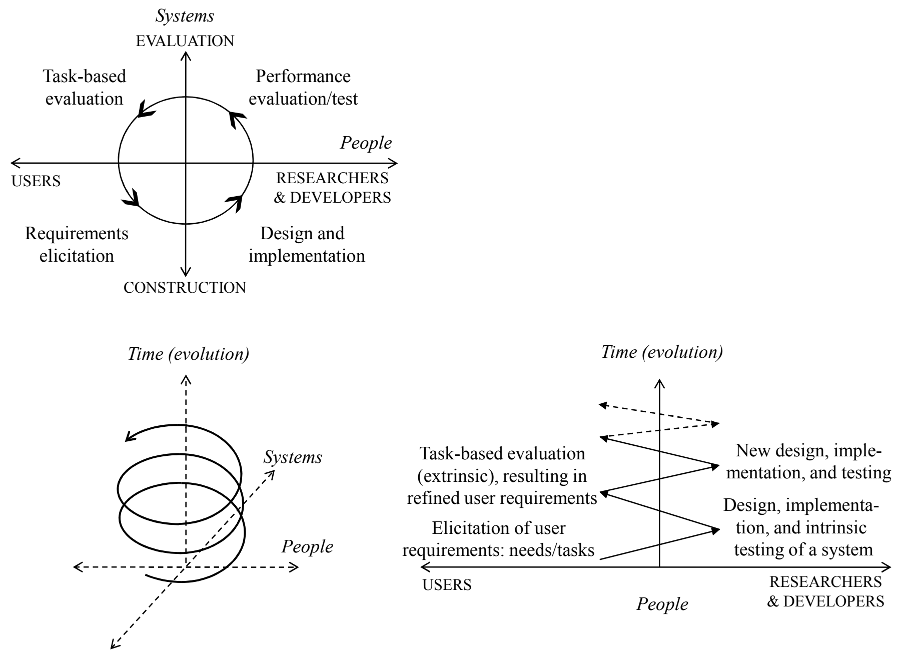

Figure 11.1: Software development process for meeting support technology: the helix model.

autonomous multimodal signal processing tools may be evaluated outside the laboratory.

In the rest of this chapter, we review the most important studies aimed at eliciting user requirements for meeting support technology, which are therefore to be situated in the bottom-left sector of the horizontal plane of the helix (visible in the upper part of Figure 11.1). Chapter 12 will present several prototypes developed mainly by the AMI Consortium (situated in the bottom-right sector of the same plane of the helix). Chapter 13 will present user-oriented evaluation methods and results (top-left sector) and will conclude by looking back on the helix model presented above.

11.3 Determining user requirements: two approaches

Studies of user needs are comparatively less frequent than specific proposals for meeting support tools, although capturing user needs normally initiates the development cycle of a software product. Two strategies have essentially been used to obtain specifications from user input. Both approaches have advantages and shortcomings, as the examples below will show.

1. The first strategy is *practice-centric* and focuses on analyzing the use of current information technology for meeting support, and inferring the needs that new technology could fulfill. This leads to firm, verifiable conclusions regarding current practices, but inferring exact specifications for future tools from them, on the grounds that they answer limitations of current ones, may require a considerable leap of faith.

2. The second strategy has been to ask users to describe functionalities that, if available to them in the future, would likely support their involvement in meetings better than existing technologies do. This requires some guidance of the polled users, outlining specifically the range of functionalities that can be expected from future technology.

The second approach is more deterministic than the first one in turning expressed user needs into precise specifications, but is still faced with a dilemma regarding generality. On the one hand, if users are left free to imagine potential functionalities, then it might be difficult to agree on a prioritized list, and many suggestions might be quite far from being possible to implement. On the other hand, if users are too constrained by feasibility issues (sometimes shown to them as a partly implemented architecture), then their answers might not reflect genuine or urgent needs. As a consequence, the resulting software might again merely reflect the designers' intuitions, with the risk of low utility or acceptance rates. Moreover, the results of the user studies are likely to mix evaluation of existing specifications with the elicitation of new ones.

11.3.1 Analysis of current practices for meeting archival and access

The purpose of a practice-centric study is to investigate and to evaluate uses of current technology. The sophistication of the current technology may vary, but the key point is to determine the capabilities and shortcomings of current practice. In this way one can first identify a clear performance baseline for further technology (new technology must at least be able to achieve what is useful in current technology) and especially identify new promising areas for research and development (areas in which new technology should overcome the limitations of current practice).

There are various ways that practice-centric studies can be carried out, but typically an ethnographic approach is used, in which the uses and shortcomings of current technology are determined through observation and interviews. In this approach, it is important that the full chain of use of current technology is analyzed. We focus in this section on studies of meeting archiving and access practices, which were the main target of the past decade, as opposed for instance to technologies for meeting enhancement or for remote meetings, which were less researched.

Two landmark ethnographic studies of practices, regarding the use of information captured from meetings in a corporate context, have been notably performed. In this studies, subjects used to held a series of project-related meetings (Jaimes et al., 2004; Whittaker et al., 2008). In each study, a dozen people were interviewed over several weeks or months, and additionally the first study surveyed 500 people, in order to explore the types of records and cues that people use to recall information from past meetings. The first study intended to explore the potential utility of visual information, while the second one (described in more detail in the following section) focused on more traditional records such as written minutes or personal notes, possibly based on transcripts of audio recordings. While both studies confirmed the importance of structured meeting minutes for recalling the information present in meetings, they differed in many other conclusions. These differences are likely due to the different perspectives of the experimenters.

In the first study (Jaimes et al., 2004), the users highlighted the utility of audio-visual recordings for verifying or better understanding points in a meeting and as an accurate overall record, while in the second one (Whittaker et al., 2008) the users emphasized the limitations of official minutes for recalling specific details – a limitation partly overcome by private notes. Searching verbatim meeting records was a potentially challenging task: the first study showed that visual cues related to the meeting room and the participants facilitated recall, as did the list of topics discussed in the meeting, while the second study put forward the difficulty to retrieve important items such as assigned tasks or decisions, and demonstrated the need for summaries rather than for full records.

Two other ethnographic studies, one by Cremers et al. (2007) and the other one by Bertini and Lalanne (2007), with respectively 10 and 118 users, confirmed the previous insights. In order to retrieve information about a past meeting they attended, people use minutes and personal notes, though almost just as often they rely on personal recollection or even on emails and their attachments. The utility of audio-visual recordings alone was considered to be quite low, the main reason – for about half of the participants of the second study – being the time that is needed to go through the recording of an entire meeting. Given this constraint, it is of no surprise that recordings were viewed as useful mainly to check what someone has said, in case of doubt, or as a proxy for people who missed an important meeting. Among the reasons why someone would need to review a past meeting, the most frequent ones are the need to remember past topics, assigned tasks, or the date of the next meeting, in order to prepare for it.

11.3.2 A practice-centric study of access to past meeting information

To illustrate the methods and possible conclusions of practice-centric studies, this section provides as an example an overview of the second study mentioned above (Whittaker et al., 2008). The authors of the study selected two service companies in the UK, one responsible for national and international mail deliveries and the other one supplying software services. In each firm the authors studied a core team of people who attended a series of meetings over a period of time. The study thus followed two teams in repeated interactions, rather than a large set of loosely connected meetings, because an important objective was to determine how information in earlier meetings was invoked and followed up on in later meetings. Both sets of meetings were oriented towards specific tasks that had to be solved and were often structured around written agendas.

Only 56% of the meetings that they observed had minutes taken. This seemed to depend on factors such as importance, meeting context and meeting type. Minutes were taken more often in the software company than the delivery company. A possible reason was that the software meetings were contractual, i.e. various promises were being made about what services would be delivered. Both parties felt that, in this case, it was advantageous for decisions and commitments to be a matter of record.

In both settings, minutes (when taken) had clear benefits in serving as a group contract and memory aid. Still, the results of the study pointed out a large variety of *limitations of public records*, obtained either explicitly from the declarations of individuals, or from observations of their behavior. Minutes appeared to be occasionally inaccurate, or even selective at times, omitting politically sensitive information; they were also laborious to produce, and thus often not timely. Minutes appeared to lack sufficient detail to allow participants to carry out personal actions, or to allow non-attendees to determine what went on in the meeting. In fact, the minutes did not capture more peripheral aspects of the meeting such as "awareness" information that is relevant to the group's functioning but not directly related to a decision or action, and did not render the individual experience of being in a specific meeting.

One response to the limits of meetings was for individuals to supplement minutes with personal notes. Indeed, when there were no minutes, participants relied on the manager's notes if these were available, or on a combination of different team members' personal notes. But, even when public minutes were made, personal notes appeared to give more detailed context and background information associated with personal and group commitments. However, participants also cited a number of *limitations of personal notes*: taking notes reduces one's ability to contribute to discussion; personal notes sometimes lack both accuracy and comprehensibility;

and they are of course even less suitable than public minutes for allowing non-attendees to understand the meeting.

To conclude, the four studies cited in section 11.3.1, and in particular the one presented in more detail in this section, indicate a clear need for meeting records, which is only imperfectly fulfilled by traditional practices of public and personal meeting notes. Moreover, raw audio-visual recordings of meetings appear to be of little use on their own, and need to be accompanied by search and browsing tools that offer finer-grained access than current media players do. Such tools should nevertheless support current record taking practices, and in particular the need for abstracting information from meetings, while addressing the problems associated with these practices.

11.3.3 Elicitation of requirements from potential users

A second approach to finding requirements for meeting support technology calls into play the users' own imagination. In this approach, requirements are elicited directly from users by encouraging them to formulate explicit technological needs, generally after the intended setting and some technical guidelines are explained to them.

This section will review several studies for requirements elicitation that follow from the practice-centric studies described above, and aim at refining the specifications of technology for meeting archiving and access. In other words, we focus here on studies of concrete proposals for addressing the limitations observed in the previous sections, specifically related to the content of meetings, rather than reviewing user studies about all aspects of meeting support technology (ranging from the arrangement of meeting rooms to means for video conferencing).

The requirements elicitation studies presented below asked participants to imagine that they are using an "intelligent" search and navigation tool for meeting recording, and to describe the tasks that it could perform, or even more specifically the queries that they would use (see also 11.4). The studies included tasks that users could perform with the help of a system, tasks that a system could be expected to perform upon receiving a command, or formal queries about meeting data – thus illustrating the variety of settings that subjects can be induced to imagine.

The instructions given to the participants have a strong influence on the elicited needs and requirements. For instance, users might express a need for tools that help them produce public minutes for a meeting, i.e. tools to help humans perform a meeting-related task. Other users might want to submit queries to a fully automatic system, such as "what were the main points discussed?" or "give me the summary of a meeting" (see Chapter 10 for current capabilities in meeting summarization). Or, they might only submit a query to find regions of interest in a meeting, which would help them produce the minutes after watching the respective parts of the meeting.

In one of the requirements elicitation studies reported by Cremers et al. (2007), eight users were asked to imagine an application generating public meeting minutes from recordings. The most demanded pieces of information to include in such minutes appeared to be the arguments for decisions, the main topics and things to do, but also simply the meeting agenda and the names of the participants. When trying specifically to catch up on a missed meeting, users emphasized the need for a summary or gist, together with a list of things to do, accompanied by a browser adapted to the visualization of the minutes in relation to a recording. In a query set with about 60 items collected from a dozen professionals by Banerjee et al. (2005), the most frequently requested item was also the list of topics or themes discussed at a meeting.

Several large sets of requirements or specific queries were collected by members of the AMI and IM2 consortia towards the beginning of the projects. One experiment included only researchers or developers of meeting technology and did not specify use cases for accessing meeting recordings (Lalanne and Sire, 2003), while another one collected "queries" as observations of interest (see next section).

Another experiment featured 14 researchers and 14 people who had not been previously exposed to meeting technology (Lisowska, 2003; Lisowska et al., 2004). The participants could choose between four use cases – a manager tracking employee performance (5 subjects) or project progress (4), an employee missing one project meeting (12) or joining an ongoing project (7) – and were asked to state in their own words the questions that they would ask to access the information in a meeting archive. About 300 queries were collected and analyzed, with the purpose of inferring requirements for meeting processing, regardless of feasibility, such as the extraction of specific features from meeting media.

11.4 Query analysis

Narrowing even further the quest for user requirements, several studies have addressed the problem of analyzing large sets of queries formulated by potential users of an hypothetical meeting analysis and retrieval system with the global goal of understanding "what happened" in a past meeting. Of course, in this case, a system for meeting retrieval must not only be able to answer the queries (e.g. find all questions raised by participant X on topic Y) but must also understand the language-based or multimodal query itself (e.g. "what did X ask about Y?"). Few studies have investigated the requirements for understanding queries[1] and we will not discuss these any further, instead we focus on the type of information that users would like to

[1]Most studies assume that the user interfaces can assist the user in formulating a complex formal (non-ambiguous) query without the need for analyzing linguistic input.

look for in a meeting archive.

The study by Lisowska et al. (2004) found that users of a meeting capture and access system would mainly look for: (1) queries related to the interaction between participants, touching on elements such as decisions, questions, discussions, or disagreement; and (2) queries about items that are conceptually part of meeting activities, such as dates, people, documents, and presentations, and also global and local discussion topics. These categories, and their sub-divisions, appeared to be overlapping by necessity, as queries can target the communicative or the content dimensions of a meeting fragment or utterance at the same time. Answering the queries requires topic detection, e.g. terms or significant keywords, named entity recognition, but also an understanding of the interaction structure, e.g. in terms of speech acts or decision processes, which in many cases far exceeds current processing capabilities. A sizable number of queries were directed towards elementary meeting items, such as presentations, agendas and dates, and can be answered using simple processing of meeting recordings. The same query set was later reused by Pallotta et al. (2007) to justify the need for argumentative analysis of meeting data – a notion loosely defined to include issues, proposals, arguments and decisions.

Query analysis can also be done on the data obtained using the Browser Evaluation Test (the BET, more fully described in Chapter 13) and more specifically using its query collection procedure (Wellner et al., 2005; Popescu-Belis et al., 2008). In the experiment reported by Wellner et al. (2005), 21 subjects were asked to formulate observations of interest regarding three recorded meetings from the AMI Corpus (see Chapter 2 of this book, Carletta (2007). The observations captured aspects that the subjects, who did not participate in the meetings, thought to have been important to the actual participants. These aspects are then considered as potential targets for subsequent search. Users were explicitly asked to mark observations as either local or global, i.e. for a given moment, a short interval, or throughout the meeting. However, the design of the collection procedure using an audio-visual meeting player encouraged observers to formulate many more local than global queries, thus possibly leading to a biased set.

In the non-consolidated set of 572 statements from 21 observers, 63% of the statements refered to specific moments, 30% to short intervals, and only 7% were about the entire meeting. As for the content, five classes can be distinguished: statements about decisions (8%), about facts stated by participants (76%, including arguments leading to decisions), and about the interaction process or the media used by participants (11%); additionally, statements about the agenda and about the date of the following meeting were infrequent (2% each) but mentioned by most subjects. If the same analysis is made over the 251 statements mentioned by at least three subjects each, then the proportions of statements regarding decisions, agenda and dates increase to 13%, 4% and 3% respectively, while those related to process/media decrease to 2% and those regarding facts or arguments remains constant.

In fact, it is important for system designers to find out not only *what* queries are asked but also *how* they are asked. For instance, Lisowska et al. (2007) conducted a large-scale study with 91 subjects, using a Wizard-of-Oz approach, hence with a partially-implemented interface, giving access to an archive of meeting recordings. The goal of the study was to observe the modalities most often used to access the archive when the subjects completed tasks assigned by experimenters, such as answering questions about one or more meetings. The study showed that exposure and training had a strong impact on the way people used modalities to formulate queries – speech, written language, or mouse clicks – with no single natural combination standing out. Speech was slightly preferred over other modalities to interact with the system, as the system appeared to understand it acceptably, thanks to the dedicated human Wizard-of-Oz in the background.

11.5 From requirements to specifications: focus on meeting browsers and meeting assistants

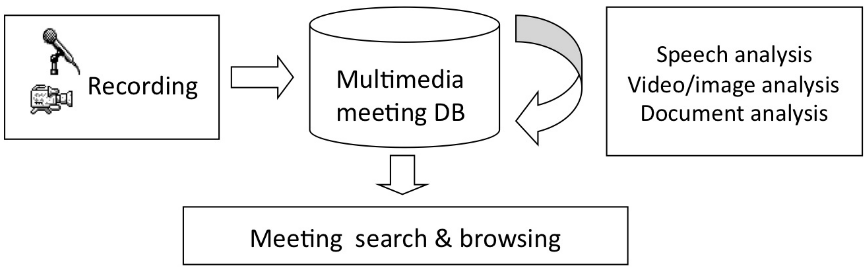

Figure 11.2: Generic architecture of a meeting processing and retrieval system: the main application of our consortium (meeting browsing for fact-finding and verification) chosen for its adequation with user needs, potential technology developments, and generality.

This review of user studies shows that requirements for meeting archiving and browsing technology are multi-faceted, but that their main dimensions are now well understood. Requirements can be categorized in terms of:

- *targeted time span* within a meeting or series of meetings, i.e. utterance, fragment, or entire meeting;
- *targeted media*, such as audio, video, documents, presentations, emails;
- *complexity of information* that is searched for, either present in the media or inferred from content;

- *query complexity or modalities used to express it.*

Two main categories of applications answer part of these requirements each: systems for meeting summarization (discussed in Chapter 10 of this book), and meeting browsers, i.e. systems for navigating meeting records and helping users to locate relevant information (which are the topic of the following chapter).

The development and testing of systems for meeting browsing and search, including fact finding and verification, has been an important focus of the AMI and IM2 Consortia. Meeting browsers answer some of the most frequently mentioned user needs, and raise interesting challenges for multimedia processing (see Chapters 3–9). Meeting browsing as illustrated in Figure 11.2 appears to be a significant transversal application, from meeting recording to storage, analysis and finally searching and browsing, striking a good balance between answering user needs, feasibility, and generality. Therefore, as outlined in this chapter, user studies from AMI and IM2 have been gradually narrowed down towards the elicitation of specific fact-finding tasks.

Still, the development of meeting capture and access systems could be made more user-driven in the future, with a number of challenges to be addressed. As user studies are notoriously difficult to generalize, a large number of studies are needed to circumscribe the range of options for meeting archiving and access technology. Such studies, however, become quickly outdated as the underlying technologies evolve continuously. Moreover, a large proportion of user studies carried out by private companies for the development of proprietary products are never published, as they offer companies a competitive advantage. This shows the need for periodical reviews or syntheses of studies related to user needs, requirements, and user-centric evaluation.

Going beyond the elicitation of requirements for meeting browsers and their related development, research and development within the AMIDA project has evolved towards co-located and remote meeting support technologies, more specifically towards meeting assistants. While meeting browsers, as addressed in the AMI and IM2 projects, are offline assistants that use the results of multimodal signal processing techniques applied to the recordings of past meetings, meeting assistants are online, and as such work in real-time to support people during meetings. This shift towards real-time support has been natural with the experience gained with offline systems. The usability engineering cycles for eliciting the requirements and designing meeting assistants could thus follow a shorter path than in the AMI project. Two main concepts for meeting assistants were thus explicit from the early stages, and once their the technical feasibility was assessed by the technology experts of the consortium, the design of the “content linking” and “engagement and floor control” systems could start (see Chapter 12, Section 12.3). In fact, user scenarios (users, tasks, context) were used to build user interface mock-ups (low-fidelity prototypes) that were shown to focus groups, which evaluated and refined the prototypes. Finally, a test-bed was developed with the most promising designs, in which the effectiveness, efficiency, and satisfaction of the meeting assistant functionalities were tested, including remote meeting support.

11.6 Summary and further reading

This chapter reviewed several user studies aimed at collecting requirements for meeting support technologies, and in particular for systems aimed at accessing recordings of meetings. Starting from a model of software development, the helix model, the chapter explained why such systems are difficult to specify completely from the very first user study. Rather, the studies were gradually narrowed down from practice-centric ones to the elicitation of specific queries that users might address to a system. Therefore, the studies did not lead to a unique specification, as they depended on how subjects were prompted to respond and how their answers were analyzed.

These requirements have lead most directly to the design and development of meeting browsers for fact finding and verification, which answer the most frequently mentioned user needs, and are described in Chapter 12. In addition, benchmarking methods grounded in user studies were designed and are presented in Chapter 13.

User requirements for real-time meeting assistants, to be used mainly during meetings, are discussed by Cremers et al. (2008), while user requirements elicitation for co-located and remote meeting support technology is discussed in the AMI Project Deliverable D6.2 (AMI Consortium, 2005) and AMIDA Project Deliverable D2.2 (Post et al., 2008), both public and containing additional references.

For more depth on the topics discussed in this chapter, readers might be interested in reading the discussion about the meeting context and its implications for system design provided by Elling (2007). Further, a summary of AMI-related achievements at various stages of the helix model appears in (Popescu-Belis et al., 2011). On a related note, a critical analysis of the extent to which current life-logging systems (a generalization of meeting capture systems) answer real user needs is provided by Sellen and Whittaker (2010).

11.7 Acknowledgments

The authors acknowledge the support of the IM2 NCCR (both authors) and the AMIDA EU project (second author). We are particularly grateful to

Agnes Lisowska-Masson for her helpful comments and careful proof-reading of an earlier version of this chapter.

Bibliography

AMI Consortium. Use cases and user requirements. Deliverable D6.2, AMI Integrated Project, April 2005. URL http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/D6_2.pdf.

Satanjeev Banerjee, Carolyn Rose, and Alexander I. Rudnicky. The necessity of a meeting recording and playback system, and the benefit of topic-level annotations to meeting browsing. In *Proceedings of INTERACT 2005 (10th IFIP TC13 International Conference on Human-Computer Interaction)*, LNCS 3585, pages 643–656, Rome, 2005.

Enrico Bertini and Denis Lalanne. Total Recall survey. Technical report, University of Fribourg, Department of Computer Science, August 2007. URL http://diuf.unifr.ch/people/lalanned/Articles/TR-survey-report0807.pdf.

Barry Boehm. A spiral model of software development and enhancement. *ACM SIGSOFT Software Engineering Notes*, 11(4):14–24, 1986.

Jean Carletta. Unleashing the killer corpus: experiences in creating the multi-everything AMI Meeting Corpus. *Language Resources and Evaluation*, 41(2):181–190, 2007.

Anita Cremers, Inge Kuijper, Peter Groenewegen, and Wilfried Post. The Project Browser: Supporting information access for a project team. In *Proceedings of HCII 2007 (12th International Conference on Human-Computer Interaction), Human Computer-Interaction, Part IV*, LNCS 4553, pages 571–580, Beijing, 2007.

Anita H. M. Cremers, Maike Duistermaat, Peter L. M. Groenewegen, and Jacomien G. M. De Jong. Making remote 'meeting hopping' work: assistance to initiate, join and leave 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 316–325. Springer-Verlag, Berlin/Heidelberg, 2008.

A. Dix, J. Finlay, G. Abowd, and R. Beale. *Human-Computer Interaction*. Prentice Hall, 1993.

Erwin Elling. Tools for fun and fruitful meetings: "don't let meetings make you go bananas!". Master's thesis, Master of Science in Human Media Interaction, University of Twente, Enschede, the Netherlands, August 2007. URL http://essay.utwente.nl/642/.

ISO/IEC. *ISO/IEC 9126-1:2001 (E) – Software Engineering – Product Quality – Part 1:Quality Model*. InternationalOrganization for Standardization / InternationalElectrotechnicalCommission, Geneva, 2001.

Alejandro Jaimes, Kengo Omura, Takeshi Nagamine, and Kazutaka Hirata. Memory cues for meeting video retrieval. In *Proceedings of CARPE 2004 (1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences)*, pages 74–85, New York, NY, 2004.

Per Kroll and Philippe Kruchten. *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Addison-Wesley Professional, Harlow, 2003.

Denis Lalanne and Stéphane Sire. Analysis of end-user requirements: sample queries. Technical report IM2.AP, IM2 NCCR (Interactive Multimodal Information Management), 2003 2003.

Agnes Lisowska. Multimodal interface design for the multimodal meeting domain: Preliminary indications from a query analysis study. Technical report IM2.MDM-11, IM2 NCCR (Interactive Multimodal Information Management), November 2003 2003.

Agnes Lisowska, Andrei Popescu-Belis, and Susan Armstrong. User query analysis for the specification and evaluation of a dialogue processing and retrieval system. In *Proceedings of LREC 2004 (4th International Conference on Language Resources and Evaluation)*, pages 993–996, Lisbon, 2004.

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 Jos17 CA, 2007.

Vincenzo Pallotta, Violeta Seretan, and Marita Ailomaa. User requirements analysis for Meeting Information Retrieval based on query elicitation. In *ACL 2007 (45th Annual Meeting of the ACL)*, pages 1008–1015, Prague, 2007.

A. Popescu-Belis, D. Lalanne, and H. Bourlard. Finding information in multimedia records of meetings. *IEEE Multimedia*, 2011.

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, 2008.

Wilfried Post, Erik Boertjes, Anita Cremers, Jacomien de Jong, Alex Jaimes, Lukas Matena, Andrei Popescu-Belis, and Simon Tucker. User requirements elicitation and interface concepts. Deliverable D2.2, AMIDA Integrated Project, April 2008. URL http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/AMIDA_D2_2.pdf.

A.J. Sellen and S. Whittaker. Beyond total capture: a constructive critique of lifelogging. *Communications of the ACM*, 53(5):70–77, 2010.

Ian Sommerville. *Software engineering*. Addison Wesley, Harlow, 9th edition, 2007.

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, 2005.

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.

Bibliography

AMI Consortium. Use cases and user requirements. Deliverable D6.2, AMI Integrated Project, April 2005. URL http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/D6_2.pdf.

Satanjeev Banerjee, Carolyn Rose, and Alexander I. Rudnicky. The necessity of a meeting recording and playback system, and the benefit of topic-level annotations to meeting browsing. In *Proceedings of INTERACT 2005 (10th IFIP TC13 International Conference on Human-Computer Interaction)*, LNCS 3585, pages 643–656, Rome, 2005.

Enrico Bertini and Denis Lalanne. Total Recall survey. Technical report, University of Fribourg, Department of Computer Science, August 2007. URL http://diuf.unifr.ch/people/lalanned/Articles/TR-survey-report0807.pdf.

Barry Boehm. A spiral model of software development and enhancement. *ACM SIGSOFT Software Engineering Notes*, 11(4):14–24, 1986.

Jean Carletta. Unleashing the killer corpus: experiences in creating the multi-everything AMI Meeting Corpus. *Language Resources and Evaluation*, 41(2):181–190, 2007.

Anita Cremers, Inge Kuijper, Peter Groenewegen, and Wilfried Post. The Project Browser: Supporting information access for a project team. In *Proceedings of HCII 2007 (12th International Conference on Human-Computer Interaction), Human Computer-Interaction, Part IV*, LNCS 4553, pages 571–580, Beijing, 2007.

Anita H. M. Cremers, Maike Duistermaat, Peter L. M. Groenewegen, and Jacomien G. M. De Jong. Making remote 'meeting hopping' work: assistance to initiate, join and leave 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 316–325. Springer-Verlag, Berlin/Heidelberg, 2008.

A. Dix, J. Finlay, G. Abowd, and R. Beale. *Human-Computer Interaction*. Prentice Hall, 1993.

Erwin Elling. Tools for fun and fruitful meetings: "don't let meetings make you go bananas!". Master's thesis, Master of Science in Human Media Interaction, University of Twente, Enschede, the Netherlands, August 2007. URL http://essay.utwente.nl/642/.

ISO/IEC. *ISO/IEC 9126-1:2001 (E) – Software Engineering – Product Quality – Part 1:Quality Model*. InternationalOrganization for Standardization / InternationalElectrotechnicalCommission, Geneva, 2001.

Alejandro Jaimes, Kengo Omura, Takeshi Nagamine, and Kazutaka Hirata. Memory cues for meeting video retrieval. In *Proceedings of CARPE 2004 (1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences)*, pages 74–85, New York, NY, 2004.

Per Kroll and Philippe Kruchten. *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. Addison-Wesley Professional, Harlow, 2003.

Denis Lalanne and Stéphane Sire. Analysis of end-user requirements: sample queries. Technical report IM2.AP, IM2 NCCR (Interactive Multimodal Information Management), 2003 2003.

Agnes Lisowska. Multimodal interface design for the multimodal meeting domain: Preliminary indications from a query analysis study. Technical report IM2.MDM-11, IM2 NCCR (Interactive Multimodal Information Management), November 2003 2003.

Agnes Lisowska, Andrei Popescu-Belis, and Susan Armstrong. User query analysis for the specification and evaluation of a dialogue processing and retrieval system. In *Proceedings of LREC 2004 (4th International Conference on Language Resources and Evaluation)*, pages 993–996, Lisbon, 2004.

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 Jos17 CA, 2007.

Vincenzo Pallotta, Violeta Seretan, and Marita Ailomaa. User requirements analysis for Meeting Information Retrieval based on query elicitation. In *ACL 2007 (45th Annual Meeting of the ACL)*, pages 1008–1015, Prague, 2007.

A. Popescu-Belis, D. Lalanne, and H. Bourlard. Finding information in multimedia records of meetings. *IEEE Multimedia*, 2011.

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, 2008.

Wilfried Post, Erik Boertjes, Anita Cremers, Jacomien de Jong, Alex Jaimes, Lukas Matena, Andrei Popescu-Belis, and Simon Tucker. User requirements elicitation and interface concepts. Deliverable D2.2, AMIDA Integrated Project, April 2008. URL http://www.amiproject.org/ami-scientific-portal/documentation/annual-reports/pdf/AMIDA_D2_2.pdf.

A.J. Sellen and S. Whittaker. Beyond total capture: a constructive critique of lifelogging. *Communications of the ACM*, 53(5):70–77, 2010.

Ian Sommerville. *Software engineering*. Addison Wesley, Harlow, 9th edition, 2007.

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, 2005.

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.