

Towards a Framework for Real Time Requirements Elicitation

Mark Gall, Bernd Bruegge
Technische Universität München
D-80290 München
bruegge@in.tum.de, gall@in.tum.de

Brian Berenbach
Siemens Corporate Research, Inc.
Princeton, NJ 08540 USA
brian.berenbach@siemens.com

Abstract

Eliciting complete and correct requirements is a major challenge in software engineering and incorrect requirements are a constant source of defects. It often happens that requirements are either recorded only partially or not at all. Also, commonly, the rationale behind the requirements is not recorded or may be recorded, but is not accessible for the developers who need this information to support the decision making process when requirements change or need clarification.

Our proposed framework is designed to solve those problems by using video to record the requirements elicitation meetings and automatically extract important stakeholder statements. Those statements are made available to the project members as video clips by using an RE database to access the statements and/or by the integration with the Sysiphus system.

1. Introduction

In traditional requirements engineering processes the requirements are defined using notes that have been created during the requirements elicitation sessions with one or more important stakeholders.

The different stakeholders that are involved in these meetings may not share a common language or project knowledge as these meetings occur in early phases of the project; misunderstandings are commonplace. The notes produced by a requirements engineer taking the role of the scribe may be incomplete, inconsistent or incorrect for several reasons, e.g. the scribe may omit important statements of a stakeholder, misinterpret a statement, only partially note a statement or even note something incorrectly.

Creating a complete, consistent and correct requirements specification out of some error prone meeting notes

is challenging and as a result the produced requirements specifications are in general also incomplete, inconsistent and incorrect. In fact as the notes undergo several transcriptions until the final requirements specification is produced, the changes may introduce even more defects. In a recent project run at a Siemens company the post-mortem defect analysis done by SCR showed that 40% of the defects in the system were due to incorrect, incomplete or not at all recorded requirements [3].

We propose a framework that is designed to improve the requirements elicitation process as well as the traceability of requirements throughout the complete life cycle of a project. To accomplish this task, requirements elicitation sessions are recorded using rich media and subsequently the stakeholder requests, that describe a requirement, are extracted and stored in short multimedia clips. Those clips are then stored in the Sysiphus database [23], where they can be linked to the extracted requirements or to UML model elements. Figure 1 illustrates the process.

This way the original stakeholder requests become part of the project knowledge base and all interpretations of requirements can be validated against the original statements, which provide more context than simple transcripts including the way of speaking and body language. This also avoids incomplete and incorrect transcripts as a result of some misunderstanding. In addition developers can easily access this knowledge whenever questions about the origin or the reasoning behind certain requirements occur.

The rest of this paper is organized as follows: Section 2 gives an overview of the related work in terms of existing systems. Section 3 introduces our taxonomy of requirements elicitation techniques and Section 4 explains some issues of statement extraction and our solutions. Hardware is discussed in Section 5 and Section 6 shows our framework architecture on a high level. Some Concerns are presented in Section 7. Conclusions are given in Section 8, followed by a short summary in Section 9 and closed with

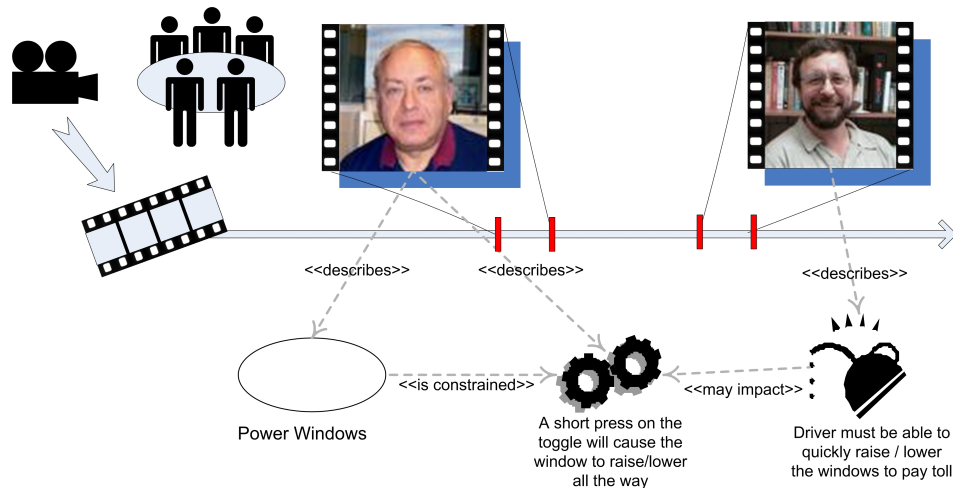


Figure 1. Statement clips associated with a use case and requirements

future work in Section 10.

2. Related Work

2.1. Multimedia Supported Meetings

Lots of research has been done in the area of multimedia supported meetings and several projects have dealt with the use of multimedia for capturing meetings for different purposes.

The meeting browser system [22] developed at CMU has provided some interesting research in tracking faces, bodies or attention during meetings as well as processing speech to track the discourse or identify the speaker. This system focuses more on the meetings as source of information, whereas we want to extract information from those video records of the meeting and focus only on those extracted parts. We also have to meet some hardware requirements, which the meeting browser system did not account for, this will be discussed in Section 5.

The Distributed Meetings system, developed by Cutler et al. [10] focuses on capturing group meetings, broadcast them to remote locations and allow people, who did not attend the meeting to watch them afterward. This system relies on the in situ infrastructure of the meeting room, which violates a requirement for our system (see Section 5).

The system developed by Chiu et al. [7] relies on a complicated camera setup, that is not practical for our purposes. The work done by Brotherton et al. [5] deals with the capturing and delivering of lectures, which follow

different structures than requirements meetings.

Additionally those systems differ significantly from the proposed framework in the way that our proposed system automatically extracts stakeholder statements from the meeting record and populates them into a requirements database.

2.2. Use of Multimedia in Requirements Engineering

Several groups have been working on the use of multimedia in requirements engineering. There has been some research on the development of new elicitation techniques such as Software Cinema [6], and integrating video into existing elicitation techniques like ethnography [15] or scenario based requirements elicitation [12].

Kaiya et al. [16] have developed a system for the support of requirements elicitation meetings, which aims to prepare agendas for follow up meetings and the final requirements specification, but considerable effort is needed to post-process the meeting record manually. In this context they also have investigated the use of speech recognition to build specification documents automatically [18] and structure records on speech act theory [21].

The AMORE system [24], developed at CMU focused on the organization of multimedia and is in fact the predecessor of the Informedia [8] system. Apart from the changes in hardware, which will be discussed later in Section 5, the system does not allow the support of mobile requirements elicitation (see Section 3), and does not automatically extract stakeholder statements from the captured meeting record to populate them into a requirements database.

3. Taxonomy of Requirements Elicitation Techniques

With our framework we want to improve the existing requirements elicitation process, i.e. support existing requirements elicitation techniques with the fewest possible modifications to the existing processes. This is important to ensure the practical use of the framework.

A plethora of different requirements elicitation techniques exist from which the requirements engineer may choose the appropriate techniques, that fits his needs. Help in deciding which techniques to use is provided by Maiden and Rugg [17], who have performed an extensive research on elicitation techniques, showing the strengths, weaknesses and conditions that must be met in order to be able to apply a technique. Hickey and Davis [14], who have interviewed some of the world's "most experienced analysts" may also provide some advice.

The existing requirements elicitation techniques differ in several ways from each other and in order to support all of the techniques it is important to understand the differences that may affect our framework. First of all, it is important whether the technique is applied during static meetings or whether it requires a certain mobility; this directly impacts the choice of hardware that is applicable to capture the requirements. We will discuss the hardware issue in more detail in Section 5.

We also must consider the form of the captured requirements, which varies strongly depending on which kind of elicitation technique is applied, e.g. in a JAD session the requirements will be captured in form of stakeholder statements whereas, when following a "blue collar" worker to gather the requirements, they may consist of him explaining what he is doing, while fulfilling his task.

A taxonomy of requirements elicitation techniques is needed and Figure 2 illustrates our taxonomy, which differs significantly from previous categorizations of requirements elicitation techniques [4]. It influences the design of the system strongly, and by clearly distinguishing parts of the system that are influenced by the taxonomy, it is possible to design the framework in a way that it is easily adjustable for the use of different elicitation techniques.

The requirements elicitation process at Siemens usually involves requirements workshops, which are categorized as office based requirements elicitation along with JAD and other elicitation techniques that include meetings of stakeholder groups. Our work focuses primarily on this category, because the support of more than one category is

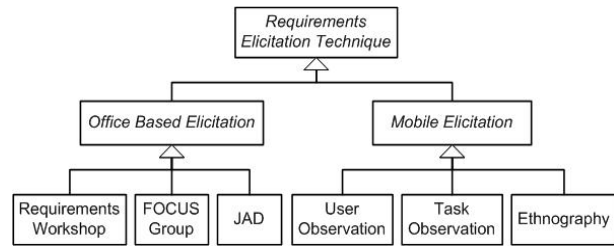


Figure 2. Requirements Meeting Taxonomy

out of the scope of this work, but the extensibility to other kinds of elicitation techniques has influenced the design of the framework to allow future support of additional techniques, e.g. user observation.

4. Statement Extraction

Current practice in office based requirements meetings is that the discussed topics are captured as notes by a requirements engineer in the role of the scribe. These notes serve as a basis for the generation of the requirements specification. The notes are seldom complete and accurate as the scribe cannot take down everything or may omit or misinterpret important statements of a stakeholder.

We aim to improve this process by using rich media to record the meetings. This way complete and accurate records of the meeting are created. The challenge is to extract important information out of the video record automatically with little or no post-processing effort. This is important for the framework to be useful in practice.

In order to achieve this goal, a structuring is needed which can be exploited for automatic extraction of important statements. Requirements elicitation meetings often have a certain structure, provided by an agenda or a presentation that is used to drive the elicitation process. Yet the discussions are sometimes most effective when they do not follow the structure. Imposing a structure to the meetings, that allows automatic extraction of the important statements and at the same time is not too intrusive to disturb the group dynamics is one of the challenges we are trying to address.

We are currently researching several approaches to this problem for the office based elicitation category. A basic structuring of a meeting without disturbing the meeting is provided by a designated kind of scribe, who uses an interface to set markings when important statements occur and may add additional information in form of annotations to the statements. The automatic generation

of most additional information is possible, but beyond the scope of this prototype and may be added in future versions.

Another approach we are currently investigating is the use of a speech recognition system, a small grammar of keywords and a trained facilitator. Using the grammar of keywords the facilitator is able to impose a basic structuring of the meeting, which is exploitable for automatic extraction of statements. This approach is more intrusive than the basic markings approach, as it may force stakeholders to repeat their statements e.g. for a summary of the elaborated requirement.

Extracting a statement from the meeting record may unfortunately remove the statement from its context, making it difficult to understand. The participants of the meeting may have spent a considerable amount of time debating, during which they developed a common understanding of the discussed topic and a basis for the comprehension of the statement.

A developer, who tracked a requirement back to a statement may lack this understanding and may need more context to fully understand the statement of the stakeholder. For that reason it is important to keep a link from the short statement clip to the meeting record, allowing the user access to the context of the statement.

5. Video Hardware

5.1. Overview

Over 10 years ago, when the AMORE system [9] was developed, storage capacity for video was one of the great challenges, typical storage devices could not handle 1 minute of a 640x480 video with a color-depth of 24-bit. Today this is not the case anymore. For example the Informedia project deals with multiple terabytes of video [13] and processes hours of video each day.

Also the capturing devices have developed. Nowadays small web cams with high resolution are available for less than 100\$. Digital cameras allow the capturing of high quality videos with sound and even cell phones with such capabilities exist. In fact a leading cell phone brand is announcing this year a new camera phone with the ability to capture 90 minutes of video in "DVD-similar" quality using only the internal memory of the phone and the storage can be further upgraded by using memory cards [1].

A system such as we propose it, is feasible with the hardware that is available today.

5.2. Selecting the Hardware

There are several requirements that impose constraints on the selection of the hardware, that is applicable for the framework. One important factor is the physical portability of the system. Although the office based category was introduced to characterize elicitation techniques that require multiple stakeholders meeting in some kind of a conference room, we cannot assume, that all requirements elicitation meetings of a project will take place in the same room or even in the same facility. The effort to transport the system from one place to the other has to be taken into account, when choosing the hardware.

Another important factor that constraints the choice of hardware is the boundary issue of setting up the system. It is impractical to use systems that produce high costs either by taking a long time to set them up or by requiring lots of people for the setup or even worse, both. It is desirable to have systems with low overhead and low personnel expenses.

Even more constraints are imposed by the mobile requirements elicitation techniques, which may require the ability to follow the "blue collar" worker without becoming an encumbrance for him. Taking into account that the requirements engineer might need to follow somebody with the capturing equipment for hours, weight and handling of the equipment also become important issues.

Another requirement that impacts the choice of the hardware, and also influences the overall system architecture is the fact, that one must assume that the requirements engineer may operate the system at the customer facilities without any access to the infrastructure of the customer, because e.g. security reasons may restrict the connection to any network for visitors. Parts of the system must therefore operate standalone.

5.3. Hardware for Office Based Elicitation

The hardware choices for capturing video in a meeting environment have been discussed by Rui et al. [20], who made the case for the use of omni-directional cameras. The benefits of such cameras are that they are less distracting to meeting participants than moving cameras, they cover the whole meeting room and they are easier to calibrate and setup than multiple cameras.

Generally there exist two kinds of omni-directional cameras, those which use a parabolic mirror as the one used by Rui et al. [20] and those that are actually arrays of low cost video cameras in a static constellation as the

FlyCam [11] or the RingCam [10]. The resolution of the video camera arrays is better than the resolution of the cameras using a curved mirror, but this results in higher storage costs for the video files. A compromise between the needed resolution and the resulting costs for storage has to be found.

Currently our framework only supports the use of a single web cam, because the support of different camera systems is out of the scope of this work and at the beginning of the development of the system no omni-directional camera was available.

6. Framework Architecture

6.1. Subsystem Decomposition

As shown in Figure 3 our system is composed of five subsystems. The capturing subsystem is responsible for the capturing of video and audio as well as the generation of meta data, e.g. speaker identity, covered topic, keyword, etc., which will be stored in form of MPEG-7 annotations. It delegates the detection of important stakeholder statements to the mapping subsystem, which allows the setting of markings and the splitting of the video and audio stream along those markings.

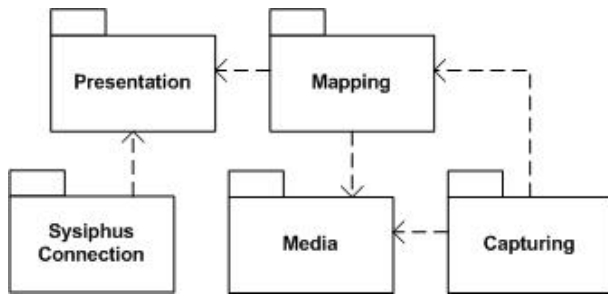


Figure 3. Subsystem Decomposition

The media subsystem provides data structures for the different kinds of media, i.e. annotations, video and audio, and interfaces for the manipulation of those structures. The presentation subsystem contains the extracted and annotated statement video clips and provides interfaces to access them. The Sysphus connection is responsible for the connection to Sysphus to enable access to the annotated statement clips from within Sysphus.

6.2. Hardware/Software Mapping

As mentioned earlier one part of the system needs to work standalone, because this part of the system will be

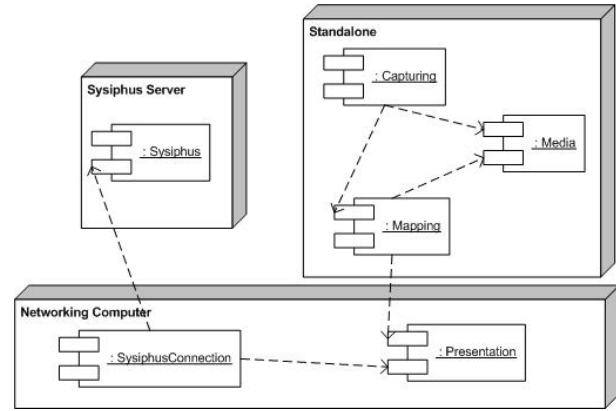


Figure 4. Hardware/Software Mapping

used at client facilities, where the access to the infrastructure may be restricted. As Figure 4 illustrates, the capturing, mapping and media subsystems need to run standalone, as this is the part that will be operating during requirements meetings.

The sysphus connection subsystem needs a network connection to the Sysphus database and can only operate from a computer that is connected through a network with Sysphus. The presentation subsystem should also reside on a networking computer, because the annotated media clips should not be stored locally on each client, but on a server to make them available for the project members.

7. Concerns

While trying to solve some problems of the requirements engineering process, we have to take care not to introduce new problems.

7.1. Filming people

A system like ours which films people, influences them simply by the fact that the people know, they are observed. Some people don't like being filmed at all and some people act differently when they are being filmed. In what way the filming alone influences the meetings and to which extent needs to be analyzed.

Cutler et al. [10] have conducted a small scale user study for their Distributed Meeting system, which uses an omni-directional camera to record meetings. The results from meeting participants indicate, that the camera setup did not get in the way of having a productive meeting,

but the participants had mixed feelings about whether they acted differently as a result of having the system recording them. In one meeting, one participant remarked, "I probably shouldn't say this because this meeting is being recorded, but..."

The question remains in which way this study is representative, whether people will get used to such a system and whether the effects will diminish after a while or whether the influence of such a system will remain constant, e.g. because people get constantly reminded that they are being recorded.

7.2. Extracting Information

Although the meeting record is inherently complete and correct, when recording a meeting using a video camera, if a person sets the markings manually, still some important information may be omitted or set out of context. We need to analyze how we can support the statement extraction in a way that as little information as possible is lost, preferably none.

8. Conclusions

The video record of the meetings provides more context than pure text transcripts. With the video record as part of the documentation, it is possible to clarify ambiguous requirements, e.g. a statement that may have been made at a meeting. In order to access the important statements of a meeting faster, the record is split into short video clips.

With the integration of the original stakeholder statements as video clips into the Sysiphus database, they become part of the project knowledge base. This way it is possible to iterate the interpretation of those statements and use them for requirements validation purposes or for illustration of the rationale behind the requirements.

In Sysiphus itself the stakeholder statements may be linked to the extracted requirements or to UML model elements, and future work may include the realization of a unified model for requirements engineering [2] in Sysiphus as Figure 1 illustrates.

Tracing the requirements back to the original stakeholder request permits a rapid answer to questions such as: "Who introduced this requirement?", "When was this requirement introduced?", "Why was the requirement introduced?", etc.

9. Summary

In this paper we have presented our proposed framework that is designed to improve the requirements engineering process in several ways. It helps during the requirements elicitation, to create correct and complete meeting records and it supports requirements validation by providing the original stakeholder statements against which the requirements can be validated.

Furthermore the framework tries to bridge the gap between the customer and the software developers by allowing the developers access to the original customer statements whenever questions about the rationale or the importance of a requirement occur. In combination with Sysiphus it is possible to trace requirements back to the original stakeholder statements.

10. Future Work

The framework is currently implemented as a proof of concept prototype. The basic capturing and splitting mechanisms have been developed, and the integration of a speech recognition system will follow.

An integration of a white board system such as the one developed by Moran et al. [19] may also follow.

References

- [1] Nokia n93. <http://www.nokiausa.com/phones/N93,7/27/2006>.
- [2] B. Berenbach. Toward a unified model for requirements engineering. In *ACSE Fourth International Workshop on Adoption-Centric Software Engineering*, pages 26–29, 2004.
- [3] B. Berenbach. Impact of organizational structure on distributed requirements engineering processes: Lessons learned. In *GSD First International Workshop on Global Software Development for the Practitioner*, 2006.
- [4] M. J. Bickerton and J. Siddiqi. The classification of requirements engineering methods. In *IEEE Symposium on Requirements Engineering*, pages 182–185. IEEE Computer Society, 1993.
- [5] J. Brotherton, J. Bhalodia, and G. Abowd. Automated capture, integration and visualization of multiple media streams. In *International Conference on Multimedia Computing and Systems*, page 54. IEEE Computer Society, 1998.
- [6] B. Bruegge, O. Creighton, and M. Purvis. Software cinema. In *CHI '04 Workshop on Identifying Gaps between HCI, Software Engineering and Design, and Boundary Objects to Bridge Them*, Apr. 2004.

- [7] P. Chiu, A. Kapuskar, S. Reuimeier, and L. Wilcox. Room with a rear view: Meeting capture in a multimedia conference room. *IEEE Multimedia*, 7(4):48–54, 2000.
- [8] M. Christel, T. Kanade, M. Mauldin, R. Reddy, S. Stevens, and H. Wactlar. Informedia digital video library. *Communications of the ACM*, 38(4):57–58, 1995.
- [9] M. G. Christel, D. P. Wood, and S. Stevens. Amore: The advanced multimedia organizer for requirements elicitation. *CMU/SEI-93-TR-12*, 1993.
- [10] R. Cutler, Y. Rui, A. Gupta, J. Cadiz, I. Tashev, L. He, A. Colburn, Z. Zhang, Z. Liu, and S. Silverberg. Distributed meetings: a meeting capture and broadcasting system. In *International Conference on Multimedia*, pages 503–512. ACM, 2002.
- [11] J. Foote and D. Kimber. Flycam: practical panoramic video. In *Eighth ACM International conference on Multimedia*. ACM, 2000.
- [12] P. Haumer, K. Pohl, and K. Weidenhaupt. Requirements elicitation and validation with real world scenes. *TSE Special Issue on Scenario Management*, 24(12):1036–1054, 1998.
- [13] A. G. Hauptmann. Lessons for the future from a decade of informedia video analysis research. In *International Conference on Image and Video Retrieval*. Springer-Verlag Berlin Heidelberg, 2005.
- [14] A. M. Hickey and A. M. Davis. Elicitation technique selection: How do experts do it? In *11th IEEE International Requirements Engineering Conference*, pages 169–178. IEEE Computer Society, 2003.
- [15] M. Jirotko, C. Heath, and P. Luff. Ethnography by video for requirements capture. In *RE*, pages 190–193. IEEE Computer Society, 1995.
- [16] H. Kaiya, M. Saeki, and K. Ochimizu. Design of a hyper media tool to support requirements elicitation meetings. In H. A. Müller and R. J. Norman, editors, *Proceedings: Seventh International Workshop on Computer-Aided Software Engineering*, pages 250–259. IEEE Computer Society, 1995.
- [17] N. A. M. Maiden and G. Rugg. Acre: selecting methods for requirements acquisition. *Software Engineering Journal*, 11(3):183–192, 1996.
- [18] N. Miura, H. Kaiya, and M. Saeki. Building the structure of specification documents from utterances of requirements elicitation meetings. In *APSEC*, page 64. IEEE Computer Society, 1995.
- [19] T. P. Moran, W. van Melle, and E. Saund. Walls at work: physical and electronic walls in the workplace. *Arbeitswelten im wandel: fit für die zukunft?*, pages 191–208, 1999.
- [20] Y. Rui, A. Gupta, and J. Cadiz. Viewing meeting captured by an omni-directional camera. In *SIGCHI Conference on Human factors in computing systems*, pages 450–457. ACM, 2001.
- [21] M. Saeki, K. Matsumura, J. Shimoda, and H. Kaiya. Structuring utterance records of requirements elicitation meetings based on speech act theory. In *ICRE*, pages 21–30, 1996.
- [22] T. Schultz, A. Waibel, M. Bett, F. Metze, Y. Pan, K. Ries, T. Schaaf, H. Soltan, M. Westphal, H. Yu, and K. Zechner. The isl meeting room system. In *Workshop on Hands-Free Speech Communication (HSC-2001)*, 2001.
- [23] T. Wolf and A. H. Dutoit. Sysphus: Combining system modeling with collaboration and rationale. *Softwaretechnik-Trends*, 24(4), November 2004.
- [24] D. P. Wood, M. G. Christel, and S. M. Stevens. A multimedia approach to requirements capture and modeling. In *First International Conference on Requirements Engineering*, pages 53–56. IEEE Computer Society, 1994.