The role of this chapter and its place with respect to the other chapters of this thesis is addressed in section 1.4 Thesis Approach and Overview.

3.1. Requirements Elicitation

Requirements elicitation is the process of obtaining the requirements of a system from users, customers and other stakeholders [IS97]. The term "elicitation" is used to point out that requirements are not simply available, ready to be captured or gathered. Stakeholders are often not aware of their actual needs at the beginning of a project or consider some requirements as obvious, which might not be the case for people unfamiliar with the field of the stakeholders. Those and further issues demonstrate the need for an engineering approach towards requirements elicitation.

3.1.1. Methodology

Various techniques are available to system analysts for requirements elicitation. An overview of the most widely used ones can be found in figure 3.1. The matrix shows the dependency between various elicitation techniques and common project conditions and characteristics. The choice of techniques heavily depends on the project at hand. Techniques, marked with "- -" are regarded as unsuitable, given the corresponding conditions on the left side of the matrix, while those that are marked with "++" are highly recommended [CR04]. Among the more straightforward factors are the project size, the project domain and the communicative abilities of the stakeholders. Project-related factors are discussed in more detail below. Other factors, such as the experience of the system analyst with the techniques, should be taken into consideration as well. Simply picking a new modern technique without considering its benefits and drawbacks might increase the risk of missing important requirements.

The project factors influencing the choice of elicitation techniques could be grouped into *human*, *organizational* and factors, which refer to the *technical content* of the requirements [CR04].

Human factors include the degree of active personal involvement of the stakeholders during requirements elicitation. Some people might have better communicative skills than other or might be better at extracting the abstract essence from concrete problems. Depending on the number of stakeholders, their expertise and the divergence/convergence of their views, some techniques might prove to be more appropriate than others in the given context.

Organizational factors might have an even higher impact than human factors. Improving an existing system might require analyzing old documentation and

Decision matrix for elicitaion techniques and project conditions

Legend: -- completely unsuitable - not well-suited o no impact => suitable + well-suited ++ very well-suited	Brainstorming	Brainstorming Paradox	6-3-5 method	Change of perspective	Walt Disney Method	Bionic / Bisociation	Osborn Checkliste	Field observation	Apprenticing	Questionnaire	Interview	Own documentation	On-Site Customer	System archeology	Reuse	Workshop	Mind Mapping	Snowcards	CRC cards	Audio recording	Video recording	Use case modeling	Essence establishment	Requirements guessing
Human factors																								
low motivation	-	--	--	--	--	--	-	+	-	+	+	--	--	o	o	+	-	+	o	o	+	o	o	++
poor communication skills	-	-	-	-	o	o	-	-	++	o	-	--	-	o	o	-	o	-	-	o	o	+	o	+
implicit knowledge	+	++	++	+	+	+	+	++	++	--	o	--	+	o	o	o	+	o	o	o	o	o	o	++
poor abstraction ability	-	--	-	-	-	--	-	++	++	+	+	-	-	o	o	o	-	-	o	o	o	-	--	+
diverging stakeholder ippinions	-	+	-	-	-	o	+	-	-	++	o	-	--	o	o	+	-	+	+	o	o	+	++	o
problematic group dynamics	--	--	--	o	--	o	+	--	++	o	o	+	+	o	o	--	-	o	-	--	--	o	o	o
Organizational conditions																								
new development	++	+	++	++	++	+	+	o	o	+	+	+	+	--	o	+	++	++	++	o	o	o	+	+
extension of an existing/old system	o	+	o	o	o	o	++	+	+	+	+	+	+	++	o	+	o	+	+	o	o	o	o	+
customer-specifc development	o	o	o	o	o	o	+	+	+	+	+	++	++	+	o	+	o	o	+	o	o	o	o	+
peoduct developoment	++	+	++	++	++	+	+	-	--	+	-	-	--	o	+	o	++	-	o	-	-	o	o	+
fixed, tight project budget	o	o	-	o	o	--	-	-	--	+	+	+	-	--	++	o	o	o	o	++	-	o	o	++
hohe Verteilung der Stakeholder	--	--	--	-	-	-	o	o	o	+	-	++	-	o	o	--	-	--	--	o	o	o	o	o
poor availability of the stakeholders	-	-	-	-	-	-	-	++	--	+	-	--	--	o	++	--	-	--	--	++	++	o	o	--
high number of stakeholders	-	-	-	o	o	-	-	-	--	+	--	--	--	o	o	--	o	--	--	-	-	+	+	o
Technical content of the requirements																								
high criticality of the system	o	o	+	+	+	-	+	+	--	+	+	+	++	++	-	+	o	+	+	++	+	o	+	-
large scale system	o	o	o	o	o	o	--	++	+	--	+	+	+	+	++	+	o	+	+	o	o	++	++	+
high complexity of the system processes	+	-	o	+	+	-	+	--	--	--	+	+	+	+	+	+	++	o	o	o	o	+	o	o
low observability	+	o	o	+	o	++	+	--	+	+	+	+	+	+	+	+	+	o	o	o	--	o	+	o
non-functional requirements	-	-	+	+	o	+	o	-	+	--	-	-	-	+	+	-	o	o	+	o	+	o	o	o
unknown field	o	o	+	o	-	-	o	+	++	--	+	++	+	++	--	+	o	o	o	o	o	+	o	--
abstract requirements	+	+	+	+	+	+	++	--	-	++	++	+	+	-	o	+	+	++	++	o	o	++	++	-
detailed requirements	-	-	-	-	-	-	o	+	++	o	++	+	++	++	++	o	-	-	-	+	+	-	o	+

Figure 3.1.: Decision matrix for elicitation techniques based on [CR04]. The techniques are classified into creative, observational, communicational, time-related and supportive tools (see the icons). Project conditions are listed on the left. The important factors for the current project and the chosen requirements elicitation techniques are highlighted.

legacy applications that are to be substituted. This would not be the case if a new application has to be developed, because there would be no artifacts to analyze. The contract relationships between client and contractor and the project budget might be crucial. For bigger projects, more developers need to be engaged. Effective mechanisms would be needed to elicit and to document the requirements in such a way that they are easily accessible to the developers as the consumers of those requirements.

The problem domain and the properties of the developed system influence also the choice of methods. Developing an embedded system might include requirements to the hardware as well, which would not be the case for a pure software project. For products of bigger size with highly complex and badly observable system workflows, a deeper analysis of the documentation and perhaps even in the source code of the system that is to be replaced would be required.

3.1.2. Choice of requirements gathering techniques

The technical context of the requirements and organizational factors affected the choice of elicitation techniques for this project the most. The main project conditions are highlighted on figure 3.1. The desire of the stakeholders to replace the existing solution with a better project cockpit that serves their needs was identified as the central factor. Both abstract and detailed requirements need to be elicited in order to meet the expectations of the stakeholders as good as possible. Project budget was an important factor, together with project duration. The availability of the stakeholders – mainly project managers and coordinators – was also limited. A high number of stakeholders was also expected. Depending on the outcome of the first iteration of requirement elicitation, the number of the stakeholders might vary dramatically. The selected techniques are described and their choice is motivated below.

System archeology is a recommended technique when developing a replacement for an existing solution. As true archeology, system archeology involves investigative work to understand the thought processes of one's predecessors [AH02]. System archeology tries to discover what the functionalities of the scrutinized artifact really are and how they fit together. System archeology intends to make sure that already present features are reflected in the new system as well, even if stakeholders take some of those features for granted and do not explicitly state them as requirements. Archeology could reveal some detailed requirements that might be otherwise easily missed or hard to obtain. The technique also helps resolving basic issues that would otherwise take time to discuss with the stakeholders. Due to those reasons, system archeology was decided to be applied in the current project.

Interviews were chosen as a further requirements elicitation technique with the intention to address those issues, which cannot be identified using system archeology. Interviews are helpful in understanding what the stakeholders are missing in the current solution and improve the chances of obtaining as many requirements as possible. They give stakeholders the chance to explain their needs in person. To further enhance the benefits of a personal interview, **audio recording** will be applied.

Questionnaires will be used, among others, to find out the central abstract requirements. They are also well-suited when the number of stakeholders is high, which was expected to be the case in the project at hand. If this expectation is met, after the first iteration of the elicitation process questionnaires will be used completely instead of interviews. As the availability of the stakeholders and the project budget and time are limited, questionnaires could help reducing the time for the interviews by providing a guideline for conducting interviews.

The chosen techniques will be combined in order to improve the overall benefits of their application.

3.1.3. The Elicitation Process: Application of the Techniques

The focus of this section is the preparation for and the application of the chosen elicitation techniques. A custom process was devised to systematize the elicitation of requirements. The process is iterative and the iterations are to be performed until the requirements are complete, understood and validated. A simplified overview of the process is shown on figure 3.2. An overview of the elicitation process is offered first. The central aspects of this process are then addressed. Note that the place of the techniques in the process and their application are discussed here. This section addresses the inputs for the system archeology, how the questionnaires were devised and how the interview sessions were prepared. The results obtained after the application of the elicitation techniques are presented in section 3.1.4.

System Archeology

System archeology was chosen as the start-off point of the elicitation process depicted in figure 3.2. The advantages, description and motivation for the choice of the technique are discussed in section 3.1.2.

GDIS (see section 1.5 Industry Partner) is currently in the process of standardization of the used cockpits. Besides the officially available Excel-based cockpit template, a multitude of home-grown cockpits are used throughout the organization. Not all representatives of the potential cockpit user groups are using cockpits.

Two cockpits, currently used at GDIS were chosen for this reason as objects of the system archeology. One is the official cockpit template, provided by the engineering process group. The other is a personal cockpit, which might give further information about the needs of its owner.

Questionnaire Creation

A universal questionnaire was developed during the first iteration. It was used, as planned in the elicitation process (see figure 3.2), during the interviews

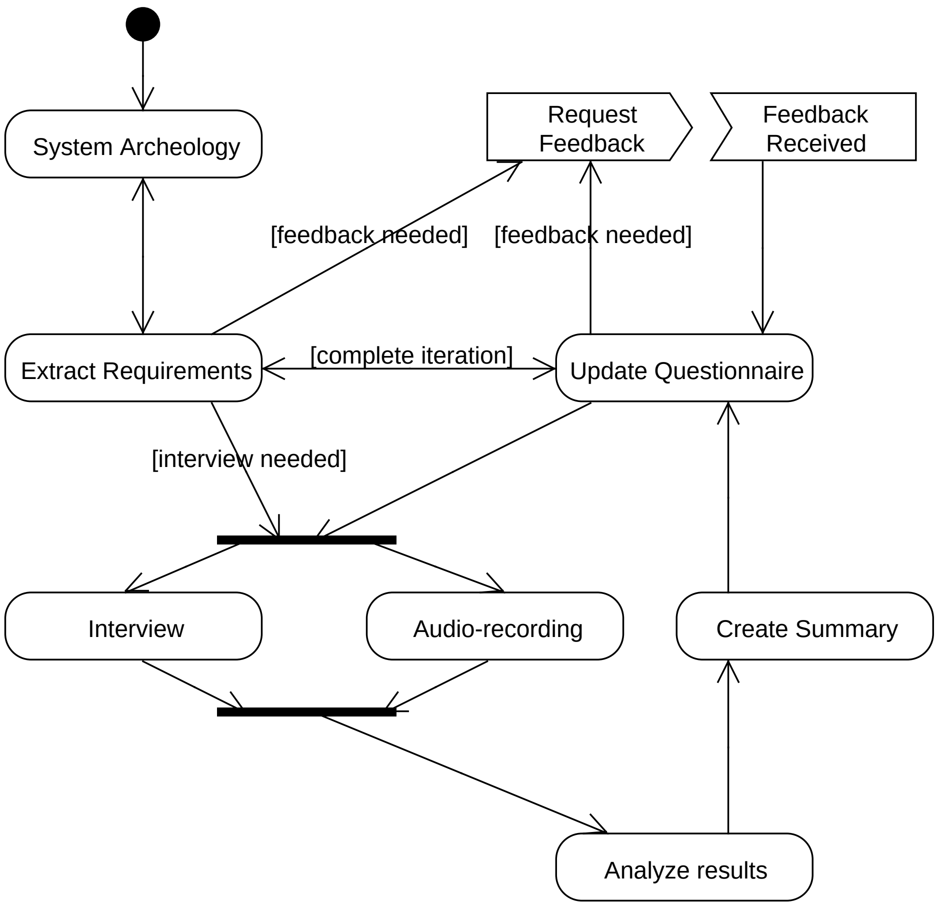

Figure 3.2.: A representation of the applied elicitation process as an activity diagram. The rounded boxes stand for activities and the arrows between them represent control flows. Bars represent the start or end of concurrent activities. Send and receive event action symbols are used for the "Request Feedback" and "Feedback Received" actions to denote possible feedback delays.

as well. The feedback of the interviewees addressed further aspects that were included in next versions of the questionnaire. A complete sample questionnaire can be found in Appendix A.

Question Types and Scale

The questionnaire contains a combination of multiple choice and open-end questions. Questions, in which the respondents were asked to rate a statement on a predefined scale, were used to evaluate the acuteness and urgency of of their needs. Likert scale was used for the latter purpose. The Likert scale is an summative accumulative rating scale, used for measuring attitudes towards given statements, available in a couple of versions. It is a useful technique to improve response rates and generalization reliability [Jup06]. The statements should express an either positive or negative attitude. In the current project, a five-level scale was used to rate the statements with the levels being "Strongly disagree", "Disagree", "Neither agree nor disagree", "Agree" and "Strongly agree". According to some methods for statistical evaluation of Likert statements, all levels of agree and disagree can be combined onto the more general categories "accept" and "reject".

Question Groups
The questions were organized into groups in order to indicate the desired aspects, under which the questions are to be answered. General questions, targeted at understanding the root problems of the current solution are asked at first. More specific questions provoke the respondents to give feedback on their usage habits and expectations towards a cockpit. The collaboration aspect and the need for interpretation assistance are evaluated. The respondents were asked to express their technology preferences as well.

Sample Population
Three groups of potential cockpit users were identified:

- project managers,
- test coordinators and
- metrics specialists.

Project managers are the main cockpit users, as identified by the system archeology. Test coordinators could also benefit from the advantages of cockpits for their daily test project operations. Test coordinators lead small test teams. Among other, they are responsible for the creation of a test plan and progress reporting. Both project managers and test coordinators need effective, proven metrics for their areas. Metrics specialists manage those metrics and observe their usage and applicability in order to drive process improvement in the organization. They are members of an organization's Engineering Procees Group (see section 1.5).

During the first iteration, a project manager and a test coordinator were interviewed. Their responses lead to the creation of differentiated questionnaires, according to the experience of the respondent with cockpits. Changes and extensions to the questionnaires had to be managed in order to provide for consistence and their easier evaluation.

Audio-recorded Interviews

As a part of the elicitation process (see figure 3.2), audio-recorded interviews were performed with one representative from each group of the sample population. Two versions of the questionnaire were used during the interviews. The questions were either read out by the interviewer or were used as guidelines for the interview. The audio records proved to be very helpful, because due to the high pace of the interviews the answers of the interviewees could not be fully written down. The records were analyzed and the answers were inserted into the questionnaire of the respective interviewee. The filled-out questionnaires were extended with questions that arose as a result of the analysis of the answers and were then sent back to the respondents for clarification and verification.

3.1.4. Elicitation results

This section presents a summary of the information, gathered through the application of the chosen elicitation techniques. The results of the system archeology are presented first, because this technique was the start-off point of the elicitation process (see figure 3.2). As discussed in the previous section, the results of the system archeology were used for the creation of the questionnaires. The information obtained with their help and with the help of the recorded interviews is presented later in this section.

Results of the system archeology

This subsection presents the results of the performed system archeology. It is sub-divided into observations, describing the analyzed cockpits, and analysis and considerations. The latter include the issues, which had to be clarified using other elicitation techniques.

Observations

The analyzed cockpits (see section 3.1.3) are Excel workbooks that contain the following information, organized in several worksheets:

- tables, representing primarily basic measures and their values over time,
- a list of the data sources,
- comments to the metrics and
- a graphical overview of the metrics.

The name of the project and the responsible project manager were located on the cover page. The cockpits were obviously intended to support single projects, as it could be seen from the information in the other sheets as well. Version history was also found on the cover page. There were two data sheets, holding weekly and monthly gathered data separately. A data source sheet contained a list of the metrics and the corresponding paths to the source files, from which the actual information is loaded. In the analyzed cockpits, those were Excel workbooks kept on a CVS server. A further worksheet contained only the diagrams, visualizing the metrics' results.

Both cockpits shared similar **metrics**. A large set of metrics measured the "Number and Status" of artifacts, such as Change Requests, Requirements and Test cases. Earned value management (EVM) – a project management technique for measuring project progress, presented in chapter 2 – was used in both cockpits to provide early warning of performance problems. Variance Analysis metrics, such as "Budget versus Plan" were also used in both cockpits. The cockpits tracked the project's CMMI compliance – the status and number of to-do items, the Process and Product Quality Assurance (PPQA) findings and an overview of the CMMI-specific practices and their degree of implementation in a project. The latter was a snapshot of an Excel table. It should be analyzed why a snapshot has been inserted in the cockpit instead of the data itself in the form of a table and whether this has been required by the cockpit users or has been a workaround. This analysis might uncover further aspects and requirements to the cockpit.

Analysis and considerations

The main **use cases** of the cockpit, as found by the system archeology, were analysis of data and reporting. An important observation of the system archeology was that the analyzed cockpits were used for reporting and not for proactive project planning and control. In the next phases of the requirements elicitation process it shall be analyzed whether the application areas of the cockpit should be extended by adding support for project planning and controlling.

The reporting chain shall also be examined to understand if there are further potential groups, who might benefit from the cockpit reports. If proved to be necessary, the information needs of the users and the currently reported data could be examined. The way of exchanging report data needs to be analyzed in detail as well. The need for collaboration within the cockpit might turn out to be a reason for the desire of the stakeholders for a new, improved solution.

At this stage of system archeology, project managers were found to be the **users** of the cockpit. Understanding the hierarchy at the company and the reporting processes could reveal further potential cockpit user roles that could benefit from a cockpit.

The overall usage of **interpretation aids** was found to be insufficient. Traffic light colors were used in one of the analyzed cockpits to denote if the values of a metric are within a desired range. Aids, such as explanations of the meaning of the graphs, maintenance and extension help could possibly improve the usability of the cockpit dramatically. Trend analysis could be applied to further ease the work of a cockpit user. Special aids for users, who just start using a cockpit for the first time, could be thought of as well. The dimensions, applicability and types of interpretation aids was decided to be discussed with the stakeholders.

Macros were used to add some **automation** to the sample cockpit, but higher degree of automation is needed. Deeper knowledge of the maintenance process of the cockpit and the plans for the future development of the environment of the cockpit are required in order to be able to analyze which data could be automatically gathered in the future.

From system archeology towards questionnaires and interviews

The performed system archeology answered questions, regarding the currently used cockpits. It also identified aspects that need to be better understood. The reasons for the need of a new, better solution have to be clarified. Ideas for improvement of the cockpits need to be evaluated by applying further elicitation techniques. The results of the system archeology were therefore applied as the foundation for the creation of the requirements elicitation questionnaire. With the help of questionnaires and personal interviews further use cases, further potential stakeholders and desired features could be discovered and issues and assumptions could be clarified.

Questionnaires and Interviews

Representatives from the different groups of the sample population that took part in the interviews or by filling of the questionnaires had different requirements. Their different levels of experience with cockpits and dashboards often notably influenced their expectations towards a possible new solution. In the following sections, the results from the evaluated questionnaires and the recorded interviews are presented, grouped into aspects.

Drawbacks of the current solution

Users, who had been using a cockpit only since recently, reported that a spreadsheet cockpit is exactly what their required currently. They had not spotted any particular problems with spreadsheet cockpits or with Excel. Not being able to access the currently used cockpit over the web was not considered a problem. It was reported that the used metrics often were project-specific or were based on individual solutions to everyday problems, which should not be the case.

Users having longer experience with cockpits shared the opinion that Excel was not the right medium for a cockpit. The main problem was the lack of centralization and the multiple issues related to migration – once the cockpits were made available to the project managers there was no longer a centralized overview of the cockpits. Project managers often customized, made changes to and extended their cockpits. That caused problems every time a new version of the cockpit was released. A change to the cockpit model required the devising of migration routines in order to assure consistency. An update in the formula for the calculation of a metric, for example, required the execution of the corresponding migration routine by every cockpit user. A solution that provided higher centralization and a common database would therefore be better.

Setting up a cockpit was considered to be another central problem. A lot of data had to be entered manually, which produced errors. Taking care of the data sources had to be done manually.

Interpretation help in the target solution

Cockpit users shared that interpretation assistance could make sense in some situations. Visual assistance, such as colors and trend analysis, was considered a very good idea by all respondents, especially if it were automatic. Traffic light marking helped a lot to obtain a quick overview of the problematic areas of a project at a glance. Trend analysis, answering the question "why is the trend like that" was considered a very good idea.

Textual interpretation aids, such as usage hints and guidances, were seen as very helpful for novice cockpit users only. For users, new to a project, to-do lists might be helpful. Such to-do lists could be used to introduce the user to the cockpit, its usage and functionalities.

As far as commenting diagrams is concerned, less experienced cockpit users expressed the apprehension that the comments would probably be standard, which would make this feature redundant. Comments were seen as supplementary to a direct personal conversation. Experienced cockpit users, on the other hand, shared the opinion that commenting whole diagrams, as well as particular points of a diagram was very reasonable. They would be helpful, because a person could usually derive many different types of information from a single diagram. Hints on the usage and maintenance were required. It should be possible to add analysis and measures in the cockpit. Commenting separate points of a diagram was also regarded a good idea. Such comments could appear for example on mouse-over.

Long-term cockpit users reported that they missed the possibility to drill-down into the data in the available solution. The ability to drill-down into data was considered a good idea, which but not for each role and only up to a certain level of detail. Each role had different information needs and different views on the data. Drill-down was only regarded as a "nice to have" feature by less experienced cockpit users, although they also noted that they could possibly turn into a requirement in the future.

Collaboration in the target solution

Collaborative work on a shared cockpit was evaluated as a good idea by the respondents. Such a cockpit would channelize the communication between project members, which was currently primarily verbal. This could not be done with Excel. In a shared solution, project members could look at the same data, but from different views and on different levels of detail. For a given role, switching the views on the same data would be helpful (aspects, information needs). Task-oriented cockpits, which guide and remind the user of tasks that need to be performed, were not considered practical. A preference for role-specific cockpits was expressed.

A cockpit, supporting collaboration would make a lot of sense, according to another respondent, because it would enable the gathering of role-specific information in an easy and convenient way. A further advantage would be the uniform way of creating metrics as well as the uniform look-and-feel of their visualization.

One of the respondents, in contrast, did not consider collaboration to be a central aspect. It strongly depended on the personal way of working, and though it could help some, it would not bring much benefits to a future solution.

Data gathering, storage and visualization

The gathering of the data could be improved in future cockpit solutions. Documents, containing much of the information required by the cockpit, should be connected to a database in such a way that data is automatically synchronized and historized. The cockpit should also offer the functionality to update the data on demand. In the latter case, one should be able to set the desired period of time. It should also be possible to go back in the history and track the changes. Oldest history could contain only aggregated data instead of all the details, although it depended on the particular case.

According to a respondent, the type of front-end, which later displays the data, had relatively low importance. A web front-end was not necessarily preferred, because GDIS use a shared server for document and data storage. A standardized Excel solution could therefore also be used for handling the user interaction. Other respondents expressed strong preferences to a web front-end. External customers were identified as a new potential group of cockpit users. A web front-end would be convenient for external customers to view a summary of the project's progress. Such interface could prove to be even more advantageous, because the usage of a cockpit could be further extended to other roles, such as configuration managers, requirements manager, chief developers, etc.

Further ideas

The target solution should be as easy to use as Excel. It should offer a choice of standard roles, which could be personalized.

The ability to import Excel tables and diagrams into the cockpit for experimental purposes was also noted as important. It would give users the possibility to devise a metric for their own purposes and test if it is reasonable or not. It makes no sense to perform a complicated database binding for a metric, which had not yet proven to be reasonable.

Another convenient feature would be the automatic generation of reports (in .pdf or .ppt format) by a single button click. This would also standardize the reports of the cockpit users.

As a further possible improvement, one could also think of a connection with Microsoft Project to visualize the project status in the cockpit.

General notes on the target solution

A high-performance role-based web solution, offering collaboration possibilities, interpretation support, would be much easier to administrate and work with as compared to an Excel solution. A participant in the evaluation referred to such a solution as *ideal*. It would enable the platform-independent creation of metrics. Additionally this application would be easily kept up-to-date.

Buying a cockpit tool was preferred to a custom built one or to the currently available Excel solutions. Costs for usage licenses, maintenance and further development had to be considered. Bigger companies were to be preferred as being more stable and reliable.

The elicited requirements are summarized and analyzed in the next section.

3.2. Requirements Analysis

The realization that *Excel is not the target solution* was an important result of the elicitation process. The stakeholders expressed their need and desire for a new improved solution.

The lack of centralization leads to high deployment times and bad serviceability. Migration routines have to be currently executed by all cockpit users every time changes to the cockpit are made. Guaranteeing that a metric is executed in all cockpits according to the same formula is difficult. Additionally, it is hard to control if and when a user has executed the necessary migration procedure. A solution to this problem is a **centralized cockpit**.

Another problem is that the cockpits work currently with operational data. They need to interact with the databases of other systems, which has many disadvantages. Operational systems can purge data and access to them might be too slow for the execution of complex queries. This poses another problem – working with operational data shifts the responsibility for versioning of data to the cockpit user. The user has to make sure that the data from potentially heterogeneous sources is correctly combined. Calculating metrics based on partially outdated information is highly undesired. A way to avoid versioning, handling complex data and inter-system dependencies is the introduction of a **data warehouse**. The differences between a standard database and a data warehouse as well as the benefits and the design of a data warehouse are discussed in section 2.4.

The need for better **collaboration** has already been recognized in the industry. The benefits of collaboration are addressed in section 2.5. Collaboration in the sense defined in this thesis could simplify the exchange of interpretations, opinions, suggestions and best practices. Sharing metrics, data and views on the data would assist the accomplishment of the common goals in a cockpit. Collaboration was overall positively rated by the survey participants. Likert scales 3.1.3 were used in the questionnaires to evaluate their attitude towards the different aspects of collaboration.

Business intelligence functionality, such as pivoting, data interaction and drill-down, etc. would make analyzing data and decision-making easier. Supporting the latter, collaboration could help for the coordination of decisions, their application and documentation.

The **integration** with other software systems needs to be improved. As noted by the respondents, adding data sources should involve as little own development as possible.

3.3. Modeling Requirements

An important step in requirements engineering is requirements validation (refer to section 3.4). A use case diagram was modeled (see figure 5.1 in section 5.1) to provide a basis for the discussion of the gathered requirements besides the summary in the previous section.

3.4. Validating and Negotiating Requirements

The analysis and an overview of the "final" version of the central requirements can be found in section 3.5. Before fixing the central requirements, they had to be validated and, where needed, negotiated.

Requirement validation ensures that the elicited requirements reflect the functionality desired by the stakeholders. It is a process, during which requirement errors are identified and fixed. Validation has a central place in requirements engineering, because fixing a requirement error after delivery may cost up to 100 times the cost of fixing an implementation error [Pun07]. Requirements validation is done to:

- check if the system supports the stakeholders' needs
- find and resolve conflicts between requirements
- make sure that no central requirements have been missed
- discuss if all the requirements can be implemented within the project constraints

The most common techniques are requirements reviews and prototyping. Informal review and feedback requests were chosen for the project at hand, based on the project specifics such as the nature of the requirements, the number of requirements in conflict, the stakeholder availability, social factors, etc.

Requirements validation proved to be of great importance, because during the validation a central issue was identified and resolved. During the process of requirements elicitation, the stakeholders explained their need for a new cockpit solution. The existing cockpits were analyzed using system archeology. The interview and questionnaire respondents expressed their need for functionality, typical for cockpits. Such functionality includes the ability to analyze project problems, describe and take measures to solve them within the cockpit. As already discussed in section 2.3, cockpits offer levers to influence the future development of a project. This functionality was evaluated positively in the elicitation process.

An important finding of the requirements review was that the stakeholders had an intention to replace the currently used Excel-based solutions stepwise. The first phase, which was already active by the beginning of the project at hand, was the standardization of the cockpits within the whole organization. The next phase was the introduction of a solution, offering primarily reporting and analytical functionality. It would be first piloted in the organization and gradually introduced to more and more of the potential user groups, identified during the elicitation phase. A product, offering extended business intelligence functionality would then be provided to the higher management and, if proven to be economical and efficient, to other user groups as well.

This additional information practically changed the weight of the separate requirements. At the first instance, the dashboard functionality (see section 2.3) would get a higher priority. The cockpit and business intelligence functionality would be of central importance for the later phases of the replacement of the currently used Excel-based solutions. The gathered requirements were analyzed from the point of view of this finding. This analysis and the resulting central requirements are the topic of the next section.

3.5. Specification of the Central Requirements

Requirement	Sub-requirement
Dasboard functionality	
	Support for data analysis
	Support for reporting
	Role-specific data views
	Customizable views
Consistency and transparency	
	Common interface for data collection/warehouse
	Centralized system to avoid migration issues
	Automatic versioning and historization
Easy maintenance	
	Integration with other IBM tools and systems
	Developed by an established, reliable company
Advanced functionality	
	Collaboration
	Multi-dimensional analysis, drill-down, data mining

Figure 3.3.: Central Requirements, an overview

The elicited requirements were discussed in the previous sections. Figure 3.3 offers an overview of the central requirements. The traditional dashboard features, such as data visualization and analysis, reporting, role-specific data views and cockpit tool customization must be supported. The tool must provide consistency and transparency by a common interface for data collection. Automatic versioning and historization data must be supported. The target product should be easy to maintain and should integrate well in the existing software infrastructure. Additionally, it should be developed by a reliable company. Support for team collaboration and advanced business intelligence features, such as drill-down, multi-dimensional analysis and data mining would be of advantage. The first three requirements and their sub-requirements have the highest priority for the first instance of the target solution. The summary presented in figure 3.3 will be used as a basis for the research and evaluation of possible solutions, presented in chapter 4 Research.

Refer also to section 1.4 Thesis Approach and Overview for an overview of the thesis regarding the logical and sequential dependencies between the applied requirements engineering process, the market research and the cockpit conception.