Select Database (SeDB) – A Database Selection Process Model

Waldemar Britts
Abstract

In recent years, new businesses opportunities have sparked the market for new types of databases and more and more companies strongly consider acquiring them. Still, however, there are many companies that use file-based, hierarchical, network and relational databases and they have no intention of transferring to some other database type. Many of them do so because of fear that promises of the new database trends may cost more than they taste. Companies may not always be aware of all the risks and problems they may encounter while transferring to a new database. Neither do they have any standard process support guiding them in this risky endeavor. Right now, there is no process model guiding organizations in a database selection process.

In this thesis, we explore the unknown domain of database selection process and suggest a database selection process model, which we call SeDB (Select Database). Our goal is twofold: (1) to create a process model to guide companies in choosing a database that is appropriate for their operation and (2) to provide a basis for further research in the domain of database selection processes.

Due to the fact that there are no published database selection process models, we had no closely related work to base our research on. We had to rely on other process works that were indirectly related to our field. Hence, we dare claim that the SeDB model was created from scratch.

The research method used was of qualitative and explorative character striving towards arriving at the following tentative hypothesis: “the suggested SeDB process model is a valid solution for addressing lack of database selection process models today”. It followed the frame of design science strongly governed by inductive reasoning. Data collection was conducted via interviews with two experts in the field, via action research and via literature study. The interviewees were selected with the convenience sampling method enhanced with a pre-defined selection criterion. Data evaluation, on the other hand, was conducted using the hermeneutics method and an evaluation model.

The evaluation model included the following evaluation criteria: (1) interviewee credibility, (2) semantic correctness, (3) syntactic correctness, (4) usefulness, and (5) process flexibility. Three rounds of interviews were conducted where the first two rounds resulted in the enhancement of the SeDB process model. In the first round, we interviewed a researcher within software engineering. In the second round, we interviewed a practitioner who was expert within database engineering. Parts of the model were also evaluated via action research. In the third round, we made a final evaluation of the enhanced SeDB process model.

The results of the evaluation proved that SeDB was useful for both the academia and industry experts. Hence, we dare claim that we arrived at a correct tentative hypothesis. We believe that the SeDB process model is a valid solution for addressing lack of database selection process models today. The model also proved to be unique in its design thanks to the new concept of activity spaces as adapted from an OMG standard called ESSENCE. Still, however, the SeDB process model needs further evaluation and extension in form of guidelines aiding companies in finding out what activities to use, when and on what organizational level.
Acknowledgments

I would like to express my gratitude to CNet Svenska AB for introducing me to the topic of a database selection process model and for letting me perform my master thesis at their company. Thanks to them, I got the opportunity to study the unexplored domain of database selection process and create something unique.

Special thanks to the CNet employees Mr. Peter Rosengren and Mr. Mathias Axling for the support they had given me as well as for the interviews they participated in. Furthermore, I would like to thank my supervisor Associate Professor Steve McKeever for the useful comments, remarks and engagement throughout the writing process. Most of all, I would like to express my appreciation for his positive and encouraging attitude towards my work.

I would like to thank the two interviewees who helped me evaluate the SeDB process model. Special thanks to the academic interviewee who put much effort into explaining many software engineering issues. Thanks also for all the ideas that I could incorporate in the SeDB process model. I should not forget to also thank the industrial interviewee who took time to learn the SeDB process model and to explain how the model’s inherent activities functioned in real life.

Finally, I would like to thank my family and friends who had to cope with “unavailable me” during the thesis writing time. Special thanks to my parents for encouraging me to study from the early primary school years till the end of my master’s education.
# Table of Content

## 1.0 Introduction
- 1.1 Problem ................................................................. 2
- 1.2 Research Question.................................................. 2
- 1.3 Purpose ............................................................... 2
- 1.4 Goals .................................................................. 2
- 1.5 Research Method .................................................... 2
- 1.6 Tentative Hypothesis ................................................. 3
- 1.7 Commissioned Work .............................................. 3
- 1.8 Bodies Involved ...................................................... 3
- 1.9 Target Audience ..................................................... 3
- 1.10 Scope and Limitations ........................................... 3
- 1.11 Terminology ........................................................ 4
- 1.12 Thesis Outline ..................................................... 5

## 2.0 Research methodology
- 2.1 Research Strategy .................................................. 7
- 2.2 Research Phases .................................................... 8
- 2.2.1 The Literature Study Phase ................................... 8
- 2.2.2 Define Evaluation Model ....................................... 9
- 2.2.3 Design and Evaluation ......................................... 9
- 2.2.4 Finalize the SeDB Process Model ......................... 10
- 2.3 Research Type ........................................................ 10
- 2.3.1 Qualitative Research ........................................... 10
- 2.3.2 Design-science Paradigm and Inductive Reasoning ... 10
- 2.3.3 Suitability for Quantitative Research .................... 12
- 2.4 Research Instruments ............................................ 12
- 2.5 Sampling Method .................................................. 13
- 2.6 Experiences Gained ............................................... 13

## 3 Databases and their Models
- 3.1 Databases and Database Models ............................... 15
- 3.1.1 Introduction to Databases .................................... 15
9.1 Conclusions ................................................................. 67
9.2 Suggestions for Future Work ........................................... 68
9.3 Epilogue ........................................................................ 69
References ........................................................................ 71
Glossary ............................................................................ 75
Appendix A List of Quality Attributes ...................................... 77
Appendix B Commissioning Companies ...................................... 79
Appendix C Activities in the ScDB Process Model ......................... 81
Appendix D Matching Terminology of Validity Threats within Quantitative and Qualitative Research ................................................................. 93
Appendix E Test Cases and Testing Results ................................. 95
Figures

Figure 2.1 Overview of our research strategy ................................................................. 7
Figure 2.2. The research phases taken in this study ...................................................... 8
Figure 2.3. The Design-science paradigm ....................................................................... 11
Figure 3.1. Definitions and illustrations of databases .................................................... 15
Figure 3.2. Illustrating traditional file-based systems ................................................... 16
Figure 3.3. Illustrating hierarchical and network data models ........................................ 17
Figure 3.4. Illustrating the relational data model .......................................................... 19
Figure 3.5. Illustrating object-oriented database models .............................................. 19
Figure 3.6. Illustrating NoSQL database models ......................................................... 20
Figure 4.1. Modified phases of process model evaluation ............................................. 27
Figure 5.1. Preliminary SeDB process model ............................................................... 34
Figure 5.2. Illustrating how process flexibility may be achieved .................................. 37
Figure 6.1. Illustrating the endeavor context and echo system of the SeDB process model 41
Figure 6.2. Overview of the SeDB process model ....................................................... 42
Figure 6.3. Groups of SeDB activity types, part 1 ....................................................... 43
Figure 6.4. Groups of SeDB activity types, part 2 ....................................................... 43
Figure 6.5. Groups of SeDB activity types, part 3 ....................................................... 44
Figure 6.6. Blueprint of the SeDB process model, part 1 ............................................. 45
Figure 6.7. Blueprint of SeDB process model, part 2 .................................................. 47
Figure 6.8. Blueprint of SeDB process model, part 3 .................................................. 48
Figure 7.1. Illustration of Comau Robot and the data it produces ................................ 53
Figure 7.2. Illustrating Output 1 and Output 2 from MongoDB and OpenTSDB. .......... 58
Figure 7.3 Old and new versions of the SDB-1 activities ............................................. 59
Figure 9.1. Operational levels and their generic responsibilities ................................. 68
Figure E.1. Output from tests 1, 2, 3, 4, 28, 52, 64 and 76 ............................................. 95
Figure E.2. Test cases fulfilling Requirements SR1 – SR2 ......................................... 96
Figure E.3. Test cases fulfilling Requirements SR3-SR4 .......................................... 97
Tables

Table 1.1 Definitions of Process, Process Model and Method .................................................. 4
Table 4.1. Evaluation criteria together with the related interview questions ............................ 29
Table 7.1. Benchmarking requirements used for database selection ...................................... 55
Table A.1. Examples of quality attributes ............................................................................. 77
Table B.1 Exploration Questionnaire ..................................................................................... 79
Table D.1 List of Validity Threats in quantitative versus qualitative research ..................... 93
Table E.1. OpenTSDB and MongoDB test results ................................................................. 98
Chapter 1. Introduction

Databases are everywhere and there are literally more than one million of them supporting various commercial and non-commercial activities. They range from file-based systems, through hierarchical, network, relational and object-oriented to NoSQL databases. Each of them fulfills a specific business goal and need. Hence, it may not always be straightforward to claim that one database type is better than the other (Berg, Seymour & Goel 2013; Burns, ud; Hellerstein, Stonebraker 2005; Sadalage, Fowler 2013).

Relational databases (RDB) have for many years governed the domain of data storage and management. Being based on the sound foundations of relational algebra, they offer elegance, data independence, simplicity and reliability to its data storage and retrieval (Elmasri, 2004 p. 21, 31, 43; Riccardi, 2003 p.8-10). At the time of their conception, neither hierarchical nor network databases could compete with this. Therefore, relational databases have spread worldwide across many businesses, and it is now hard to find any company that is not using a relational database. (Elmasri, 2004, p.21; Manoj, 2014; Sadalage, Fowler 2013 p. 3-12)

Recently, new needs have arisen requiring more data volume storage, better performance in form of higher retrieval frequency and processing, higher scalability and better support for agility. Unfortunately, relational databases cannot satisfy many of those needs. They are not well equipped for handling large complex sets of both structured and unstructured data in an efficient and cost-effective manner. Neither are they fast nor scalable enough (Manoj, 2014; Sadalage, Fowler 2013, p. 5-7; Tiwari, 2011). For this reason, the database community is searching for solutions that better accommodate to new business needs. Some of such solutions are NoSQL databases, the databases that promise to solve performance and scalability problems by storing and retrieving big data in a non-tabular or semi-tabular forms. (Manoj 2014; Sadalage, Fowler 2013, p. 3-12; Tauro, Clarence & Aravindh 2012; Tiwari, 2011)

In today’s database market and research, all types of databases (file-based, navigational, relational, traversal and NoSQL databases) are still in use and there are many of them. They are all suited to various organizational needs and businesses. Some are popular in administrative applications, some other are used to support data management in operating systems or design systems while others gain ground in the domain of Internet of Things. Common to them all is the fact that they all gather and process data that is increasing in an exponential way. In the year of 2014, research has shown that 90 percent of all the world’s data got digitalized only within the last two years (IBM 2015, Science Daily 2013). The data are not only going to expand in their size but also in their structuredness and unstructuredness, the range of their formats and the range of businesses that they are going to support.

In recent years, new businesses opportunities have sparked the market for new types of databases and more and more companies strongly consider acquiring them. Still, however, there are many companies that use file-based, hierarchical, network and relational databases and they have no intention of transferring to some other database type. Many of them do so because of the fear that promises of the new database trend may cost more than they taste.

To choose the right database is not an easy task today. A wrong choice may imply serious consequences in form of high costs, operational disturbance, annoyed staff, dissatisfied customers, and, at its worse, loss of business (Manoj 2014; Tiwari 2011). For this reason, many companies face a
critical moment of making decisions on whether to transfer to a new database technology or whether to stay with the old one. If they choose to transfer to the new technology, then they may have to pay a high price for choosing the wrong database. They may be forced to roll back to the old database or they may encounter the cost of having to choose yet another database. If they choose to stay with the old technology, on the other hand, then they may run the risk of either getting out of business within the nearest period of time, or, they may save their business by not taking on any technological risks. 

*Summing up, success of future business to be based on the choice of databases is for many companies difficult to predict.*

1.1 Problem

Today’s database market offers more than several hundreds of databases, all of them varying in data models, usage, performance, concurrency, scalability, security and the amount of supplier support provided. To stay competitive, many companies have to choose a database technology that is appropriate for their business operation (Mombrea, 2014). For this, they need guidelines helping them define a database selection process model. Unfortunately, there are no such guidelines today. To the knowledge of the author of this thesis, there is no single process model aiding companies in this very critical and important endeavor.

1.2 Research Question

All types of research should be guided with well-formulated and clearly focused research questions. To address the problem of lack of database selection process models, we specify the following research question:

*What activities should be included in a database selection process model and how should they be organized?*

This research question will guide our overall research process and devise efficient research strategy.

1.3 Purpose

The purpose of this research is to create a process model listing a set of activities that need to be performed when selecting a database. The process model is called *Select Database* and its acronym is SeDB. The SeDB process model deals with the selection of any database technology ranging from file-based systems, to hierarchical, network, relational and traversal databases, to finally, NoSQL databases.

1.4 Goals

The short-term goal of this thesis is to guide companies in choosing a database that is appropriate for their operation. The long-term goal is to provide a basis for further research in the domain of database selection process models.

1.5 Research Method

The research method used in this study was qualitative implying that it focused on exploring the process and theory associated with acquiring new databases. It followed the frame of design science (Johannesson, Perjons 2012). Being inductive and explorative in its nature (Hevner, et al. 2004), it helped us gather information about the activities that were pertinent for the database selection process. Our study was, however, enriched with the elements of action research which we used to evaluate the latter part of the SeDB process model. By following our model, we actively observed and influenced the outcome of selecting a database.
Summing up, data collection was conducted via interviews, via action research and via literature study. The interviewees were selected with the convenience sampling method enhanced with a pre-defined selection criterion. Data evaluation, on the other hand, was conducted using the hermeneutics method and an evaluation model. More information about our research method is provided in Chapter 2.

1.6 Tentative hypothesis
This research has not only resulted in the SeDB process model but also in a tentative hypothesis reading in the following: “the suggested SeDB process model is a valid solution for addressing lack of database selection process models today”.

1.7 Commissioned work
One of the companies that is in the process of transitioning to a new database technology (NoSQL) is Comau Robotics, the company that manufactures robots for automotive production (Comau, 2015). During a production process, these robots produce much data which, in turn, are used for real time production monitoring and for analyzing the manufacturing process. Right now, Comau is in great need of an appropriate database technology which might assist them in this very complex task. Therefore, they commissioned CNet Svenska AB (CNet, 2015) to analyze and choose an appropriate database technology for managing their automotive production. CNet, in turn, commissioned us to help Comau Robotics with their choice. The descriptions of the commissioning companies are presented in Appendix B.

1.8 Bodies Involved
In addition to the commissioning companies, CNet Svenska AB and Comau Robotics, two people were involved in this study. These were experts within software and database engineering. The first expert came from the academia whereas the second one came from the industry. Their role was to evaluate the SeDB process model from their respective perspectives.

1.9 Target Audience
This thesis has two main target audiences. These are the industry and academia. Regarding the industry, the primary target audience is any company, software and/or non-software that is in need of a process model guiding them in a selection of a database that is appropriate for their operation. Regarding the academia, the SeDB process model proposed in this research can be used both within software and database engineering research and education due to the fact that there are no database selection process models whatsoever. Therefore, it can provide a basis for further research within the subject and provide platform for creating educational material.

1.10 Scope and Limitations
This study results in the SeDB process model that guides companies in their selection of a database. In reality, a database selection process is an initial part of an overall database lifecycle management process. This initial part proceeds with the installation of the selected database and adaptation of its hardware and software environments. This thesis focuses only on the initial selection process and does not encompass the installation and adaptation steps.

Although our work was commissioned, we could not evaluate the initial phases of the SeDB model at the commissioning companies. Those companies could neither offer us any access to their database selection process models nor access to any professional involved in a database selection process. Instead, we had to design the SeDB process model on our own using the findings of related works in
Table 1.1 Definitions of Process, Process Model and Method

| **Process**: A set of tasks or activities performed to achieve a given purpose or a specified result. (IEEE Computer Society, 1991 & Sommerville, 2007) |
| **Process Model**: A description for how to perform a process. (Sommerville, 2007) |
| **Method**: A set of system model descriptions, rules applying to system models, recommendations for how to design a system model and process guidance describing the process activities and their organization. (Merriam-Webster, u.d & Sommerville, 2007) |

the literature studied and we had to evaluate it by interviewing professionals outside the commissioning companies.

The latter activities of the SeDB process model deal with the actual evaluation of the databases that have been chosen as candidates for the final database selection. We evaluated these activities via action research during which we actually benchmarked the candidate databases. Due to time restrictions, however, we could only focus on two databases. Here, we focused on only NoSQL databases that we deemed appropriate for the manufacturing sector. This should however not exclude the use of the SeDB in the context of choosing any other database technology.

1.11 Terminology

Most of the terminology used in this thesis is self-explanatory. However, we used some of the terms that might not be easily understood. Those terms are marked with an asterisk in the thesis body text to indicate that they are put into the glossary to be found at the end of this thesis. Some of those terms, however, already need an explanation herein. These are process, process model, and method. They are often viewed interchangeably and many times they are understood differently. To avoid confusion, we briefly explain them using the definitions in Table 1.1, the definitions corresponding to a compiled version of the current definitions as suggested by (Sommerville, 2007; Merriam-Webster, u.d.; IEEE Computer Society 1991).

As shown in Table 1.1, a process consists of a set of activities to be performed for achieving a given purpose or a specific result. Often, a process follows some process model, a description of how it can be performed. In the least, a process model includes a set of activities and the order among them. It may also include guidelines for selecting process activities and suggestions for roles performing them.

There also exists a third term – method. This term is understood differently, and many times, it is used as a synonym to a process model. Table 1.1 shows how we understand it by displaying the definition of Sommerville (2007). A method comprises a set of system model descriptions, rules applying to system models, recommendations for how to design a system model and process guidance describing process activities and their organization.

In this thesis, we are going to use the terms **process** and **process model** as defined in Table 1.1. We are not going to use the term method due to the fact that a method encompasses much more than a process model. Only the part dealing process guidelines corresponds to a process model. The other parts, the parts describing system models, rules for those system models and recommendations for achieving a good system design, are outside the scope of a process model and of this thesis as well (Sommerville, 2007).
1.12 Thesis outline
The remainder of this thesis consists of the following chapters:

Chapter 2: Research Methodology: This chapter describes the research method. It describes the research strategy chosen, research phases, respondents and the research instruments.

Chapter 3: Databases and their Models: This chapter presents various types of databases and their models. It also describes the steps of the processes within the related domains.

Chapter 4: Evaluation Model: This chapter presents the evaluation model to be used for evaluating the SeDB process model.

Chapter 5: Preliminary SeDB Process Model and its First Round Evaluation: This chapter outlines the preliminary version of the SeDB process model and its academic evaluation.

Chapter 6: Improved SeDB Process Model and its Second Round Evaluation: This chapter describes the improved version of the SeDB process model and its industrial evaluation.

Chapter 7: Benchmarking: This chapter explains the benchmarking process, the process that is part of the evaluation of the SeDB process model via action research.

Chapter 8: Analysis and Discussion: This chapter discusses and analyzes the results as collected during interviews and action research. It also explains how the validity threats were attended to and how the ethical rules were paid heed to.

Chapter 9: Conclusions and Future Work: This chapter provides final remarks and makes suggestions for future work.
2. Research Methodology

This chapter presents the research methodology taken in this study. First, Section 2.1 gives an overview of our research strategy. Section 2.2 lists all the research phases while Section 2.3 describes and motivates the type of research and choice of methods that were appropriate for this study. Section 2.4 presents the research instruments that were used for data collection and evaluation of our research results. Finally, Sections 2.5 and 2.6 describe our sampling method and experiences gained during this study.

2.1 Research Strategy

To suggest an industrially viable database selection process model was not an easy task for the author of this thesis being only a master student. It required much courage and effort. The effort got even more complicated due to the fact that there were neither any published database selection process models nor was it easy to find experts within the domain. To achieve solid and credible results and to address our research question, we had to design an appropriate strategy that might give us maximal output within the time slot assigned for writing this thesis.

Our research strategy is presented in Figure 2.1. As shown there, the whole strategy is based on the design science paradigm that is governed by inductive reasoning. It includes the following components: (1) design of research phases, (2) choice research methods, (3) method for selecting respondents, (4) construction of research instruments, (5) management of validity threats, and (6) consideration of ethical requirements. All but the last two of those components are described in this chapter. Regarding the validity threats and ethical requirements, they are presented and motivated for in Chapter 8.

![Figure 2.1. Overview of our research strategy](image-url)
Figure 2.2. The research phases taken in this study

2.1 Research Phases
This section provides an overview of the overall research process by presenting and explaining the research phases. As illustrated in Figure 2.2, these are (1) Literature Study, (2) Define Evaluation Model, (3) Design and Evaluation and (4) Finalize SeDB. They are described in Sections 2.2.1-2.2.4, respectively.

2.2.1 The Literature Study Phase
The literature study phase focused on studying databases and database theory and on studying process models that might aid us in designing the SeDB model. It consisted of three sub-phases: (1) Study of Databases and Database Theory, (2) Study of Process Models and (3) Study of Evaluation Models. As shown in Figure 2.2, the Study of Databases and Database Theory sub-phase was conducted in the initial phase of our research whereas the Study of Process Models sub-phase was continuously performed throughout almost the whole research process. Finally, the Study of Evaluation Models was conducted in parallel with the Define Evaluation Model phase to be presented in Section 2.2.2.

Finding literature dealing with databases and database theory was easy and straightforward. However, understanding the theory behind all the types of databases was very demanding. We studied all types of databases ranging from file-based systems to navigational, relational, object-oriented to NoSQL ones. All these databases differed with respect to their data models and usage. While studying them, we tried to find out whether the choice of a specific data model would impact the design of the SeDB process model. The SeDB process model should neither be inhibited by the new NoSQL data models nor hierarchical or relational data models, nor any other data model.

Finding literature dealing with database selection process models was very challenging. Here, we practically found nothing although we used a wide range of keywords such as database selection process, database selection guidelines, database selection method, database select and database framework method. We searched in research databases such as IEEE Xplore (IEEE, 2015), ACM (ACM, 2015), John Wiley & Sons (2015) and Springer (2015).

We continued our search for similar process models using keywords such as legacy system process, software system selection, software system benchmarking and software system method guidelines. The models we found dealt with legacy system migration, selection of expert system applications and benchmarking XML Database Implementations.
As shown in Figure 2.2, the literature study of the database selection process continued throughout almost all our research process and was conducted in parallel with the latter research phases. This is because while conducting the latter phases such as design of the model and benchmarking of the databases, we grew in our understanding of what was needed to be studied.

All research suggestions must be evaluated in some way. Usually, researchers identify a set of criteria against which they evaluate their research results. For this reason, in the Study of Evaluation Models, we searched for criteria that were pertinent for evaluating process models. While doing it, we realized that there were not so many research suggestions for how to evaluate suggested models before their implementation. We however found works of Sedera, Rosemann & Gable (2002) and of (Doll & Torkzadeh, 1988. It is on those works that we base the creation of our evaluation model to be used for evaluating the SeDB process model in the Define Evaluation Model phase.

2.2.2 Define Evaluation Model

Parts of the initial Literature Study led to the second phase titled Define Evaluation Model. In this phase, we outlined a model including criteria that were appropriate for evaluating the SeDB process model. The evaluation criteria were (1) interviewee credibility, (2) semantic correctness, (3) syntactic correctness, (4) usefulness, and (5) process flexibility.

Our model was evaluated via two types of evaluations: (1) evaluations via interviews and (2) evaluations via action research. The interview questions strictly followed all the five evaluation criteria. They are presented in Chapter 4. During action research, on the other hand, we only evaluated three evaluation criteria. Those were semantic correctness, syntactic correctness, and usefulness.

2.2.3 Design and Evaluation

Initially, the Design and Evaluation phase consisted of five sub-phases. These were (1) Design SeDB Round 1, (2) Evaluate SeDB Round 1, (3) Design SeDB Round 2, (4) Evaluate SeDB Round 2, and (5) Benchmark Databases. However, at the end of this phase, we created an additional phase on the fly which we called Evaluate SeDB Round 3.

In the first sub-phase, (1) Design SeDB Round 1, we outlined a preliminary version of the SeDB process model based on the literature study. While outlining the model, we also evaluated it from the modeler’s perspective according to the principle as defined in Section 4.1. In the second sub-phase, Evaluate SeDB Round 1, we evaluated it with an academic interviewee. On purpose, we first chose an academic evaluation. We believed that an academic evaluation would provide us with solid feedback on the overall design of the process model. In this phase, one academic expert was interviewed using the questionnaire as proposed in the evaluation model in Chapter 4.

In the third sub-phase, Design SeDB Round 2, we took into account the feedback received from the academic evaluation. Using it, we improved the SeDB process model and again evaluated it from the modeler’s perspective. The improved model was, in turn, used for the industrial evaluation in the next sub-phase, Evaluate SeDB Round 2. Here, we interviewed one professional who worked in the field of software and database engineering.

In the last sub-phase, Benchmark Databases, we executed some of the SeDB process activities during action research. Here, we made a real life selection of the candidate databases to be suggested to the commissioning companies. This phase allowed us to test parts of the SeDB model in a real life industrial setting.
At the end of our evaluation process, we felt urgency to conduct the third round of interviews. During the first two rounds, our interviewees were provided different versions of the SeDB process model. We felt that we needed to put them on the same level by providing them with the same versions of the model to evaluate. Therefore, we created the Evaluate SeDB Round 3 on the fly. Here, we requested from our two interviewees to evaluate the initial and improved versions of the SeDB process model and express their opinion whether the improved version was really an improvement.

2.2.4 Finalize the SeDB Process Model
In the final phase, Finalize the SeDB process model, we took into account the results gathered from the evaluation phases with the academic and industry experts together with the experiences gained during action research. Those results contributed to the final version of the SeDB process model and to the confirmation that we had arrived at a correct tentative hypothesis.

2.3 Research Type
This section presents and motivates type of research taken in this study. Section 2.3.1 presents the research type. Section 2.3.2 places our research in the framework of design research paradigm. Finally, Section 2.3.3 motivated why this study was not of a quantitative character.

2.3.1. Qualitative Research
Overall, the research type as performed in this study was qualitative implying that it focused on exploring an imaginable phenomenon that required approval of human subjects (Oates, 2008). Its aim was to acquire a deep understanding and knowledge about database selection processes and reasons that governed the choices of their inherent activities. By exploring the domain studied, we had access to a large amount of data which we then had to analyze from the what, who, how, were, when and why perspectives.

Being of a qualitative type, our research was interpretative by nature. Its aim was to investigate and analyze the unexplored and unstructured domain of databases selection processes. It included typical qualitative research methods for collecting and analyzing data (Oates, 2008; Johannesson & Perjons, 2012, Oates, 2005). Data collection was conducted via interviews with two experts in the field, via action research and via literature study. Data evaluation, on the other hand, was conducted using the hermeneutics method and an evaluation model.

The data collection method via interviews used open-ended questionnaire. The data analysis method was mainly conducted via action research and hermeneutics implying that we triangulated the results from more than one sources and methods. The sources used were literature studies, interviews and action research. By using a variety of data collection and analysis methods on one and the same topic, we could assure the validity of our research results. Hermeneutics and triangulation allowed us to cross-validate data and capture and evaluate various dimensions of the database selection process model coming from various sources.

2.3.2. Design-science Paradigm and Inductive Reasoning
Our research strictly followed the design-science paradigm that was governed by inductive reasoning along its way. Design science is a paradigm constituting a template for defining research strategies and research methods (Johanesson & Perjons, 2012). Its activities are illustrated by the rectangular boxes in the middle part of Figure 2.3. They range from explicating a research problem to outlining and designing an artefact, to then demonstrating and evaluating it. The results of these activities contribute to knowledge base building of the domain studied (the lower part of Figure 2.3). The knowledge base is therefore successively developed during the research process.
The design research paradigm only outlines the phases to be followed. It is then up to individual researchers to define their own strategies and choose appropriate methods. As indicated by Figure 2.1, we chose several research methods that we felt were pertinent for our work. However, our overall research was governed by inductive reasoning that was part of qualitative studies. Inductive reasoning fitted into the design-science paradigm in an excellent way. Its phases are illustrated with cloud symbols in Figure 2.2.

Regarding the design research paradigm, our research process followed its template in the following way:

- **Explicate problem** corresponded to our Literature Study phase during which we identified the problem of lack of process models for selecting a database.
- **Outline Artefact and Define Requirements** corresponded to our two phases: (1) the Define Evaluation Model phase during which we defined the criteria for evaluating the SeDB process model, and (2) the initial part of the Design SeDB Round 1 phase during which we defined requirements for the SeDB process model to be used for outlining its preliminary version.
- **Design and Develop** corresponded to the second part of our Design and Evaluation phase during which we finalized the preliminary version of the SeDB process model.
- The last two activities Demonstrate Artefact and Evaluate Artefact had their correspondences in three consecutive activities. Demonstrate Artefact and part of Evaluate Artefact corresponded to the Evaluate SeDB Round 1 and Evaluate SeDB Round 2 phases. Here we demonstrated the SeDB model to our interviewees and interviewed them. Another part of Evaluate Artefact corresponded to the Benchmark Databases phase during which we evaluated part of the SeDB process model via action research and to the Finalize the SeDB Process Model phase during which we used feedback.
of the interviewees and feedback from the action research for improving and finalizing the SeDB process model.

Regarding the inductive reasoning, its phases were covered in the following way:

- **Observation** during which we collected and examined specific process facts within the database selection domain. This was conducted during the Literature Study phase.
- **Pattern identification** during which we detected consistent and recurring characteristics of the process. This was performed during the Define Evaluation Model and Design and Evaluation phases.
- **Tentative hypothesis formulation** during which we further explored the identified patterns and using our evaluation criteria, we formulated a tentative hypothesis. This was conducted in the Finalize SeDB phase.
- **Theory generation** during which we examined the proposed SeDB model, identified improvements and established a foundation for future work. This was conducted in the Finalize SeDB phase as well.

### 2.3.3. Suitability for Quantitative Research

This study was not suitable for quantitative research. The goal of quantitative research is to analyze mathematical and statistical data to answer a hypothesis or a question. In quantitative methods, the researcher aims to answer questions like *how many* and with what *statistical significance*.

Quantitative methods have one big flaw in the context of explorative studies. They do not provide the researcher with enough understanding and interpretations of the research subject. Because of this, quantitative research is best suited for experiments, simulations and statistical inferences (Oates, 2008). Since this study did not aim at proving any hypothesis, quantitative methods were deemed irrelevant. The aim of this study was to explore the process and theory associated with acquiring new databases and to suggest a tentative hypothesis. Because of this, a qualitative method was chosen.

### 2.4. Research instruments

The research instruments chosen for this study were (1) research literature, (2) SeDB evaluation model, (3) SeDB questionnaire, and (4) database benchmarking criteria. Regarding research literature, it constituted an essential instrument in this study. We could neither conduct any case study nor any observation of any database selection process. Starting our work completely from scratch, we had to rely on the published work within the related domains in the initial phases of our research. We could then further evaluate our work using instruments such as the SeDB evaluation model, interviews and benchmarking criteria.

The SeDB evaluation model consisted of criteria to be evaluated with the aid of answers to the questions included in the SeDB questionnaire. Those are presented in detail in Chapter 4.2. Regarding the interview questionnaire, it included open-ended and semi-structured questions (Oates, 2008). The questions let the interviewees to provide feedback on the SeDB process model and fully share their knowledge and insight without any restraints. In this way, we could elicit new knowledge about the database selection process, however, still be able to compare and analyze the answers, and thereby, minimize the variation among the answers of our interviewees (Ibid.).

All of the interviews were conducted face to face. The interviews were recorded and transcribed. A total of three sets of interviews were conducted. The interviews were categorized to either (1) academic or (2) industrial ones depending on the interviewees’ background.
Regarding the benchmarking criteria, we used a set of requirements to be fulfilled by the databases to be benchmarked. Those requirements were stated by the commissioning companies. We also defined a set of additional requirements for selecting databases to be benchmarked. Both sets were used as an instrument for conducting the Benchmark Database phase and for using the latter phases of the ScDB process model in a real life database selection process. The two sets of requirements are described in detail in Sections 7.2.1 and 7.2.3, respectively.

2.5. Sampling method
The choice of the interviewees was based on a convenience sampling method. Other terms for this method are “non-probability sampling method” or “purposive or judgemental sampling method” (Denzin & Lincoln, 2005; Marshall, 1996). This means that the respondents were selected thanks to their convenient accessibility within the research subject. A pre-defined selection criterion was used to ensure that the right respondents were chosen. It stated that the respondents had to have had at least ten-year experience within the software engineering discipline and had been involved in creating or performing a database selection process or a similar process. In this way, we could make sure that the suitable respondents got chosen.

2.6. Experiences Gained
During our research, we encountered various obstacles that hampered our research process. Some of them could not be foreseen. For this reason, the work took much more effort than expected. The initial idea for the thesis was to evaluate NoSQL databases within the field of the Internet of Things (IoT). The evaluation of these databases would then lead to a complete process model for acquiring and implementing a new database technology. However, the limited time was not sufficient for conducting the original idea. After initiating our work, we soon realized that the scope of acquiring and implementing a new database was too huge for a master thesis. Therefore, we limited the scope of this research to only the database selection phase.

Even if we limited the scope of the thesis, we still had the problem of finding work on which we could base our research. Finding experts to be interviewed was another challenge. It took us much time not only to find knowledgeable individuals but also to convince them to support our research.
Chapter 3. Databases and their models

This chapter provides a theoretical background on which we based our research. Section 3.1 gives an account of databases and data models whereas Section 3.2 reports on the status of database selection process models.

3.1 Databases and Database Models

In this section, we provide an overview of databases. Section 3.1.1 makes a brief introduction to the concept of databases. Section 3.1.2 lists types of databases that are in use today. Sections 3.1.3-3.1.5 describes the pre-relational, relational and post-relational databases, respectively.

3.1.1 Introduction to Databases

In today’s world, databases are such an integral part of our day-to-day lives so that we are not even aware that we are using them. The term database connotes to different things depending on the context it is used in. There are two main contexts, computing and non-computing. Some of its definitions are presented on the left hand side of Figure 3.1. There, the first three definitions are primarily used outside the computing context. The mainly state that databases are collections of related data. In the computing context, as stated in Definition 4, these collections are organized according to a data model (schema) and manipulated by programs such as, for instance, a database management system. Because of this strong coupling between those three, the term database is therefore often used to connote a database, data model and database management system (Hellerstein, Stonebraker 2005).

The data is usually managed by a database management system (DBMS), a software application that interacts with one or several databases and user applications. As illustrated on the right hand side of Figure 3.1, DBMS is located between the user applications and the database. All access to a database from a user application is handled via the DBMS. In addition, DBMS provides various controls such as security, recovery, concurrency and the like (Hellerstein, Stonebraker 2005).

The database data is organized according to some data model, a conceptual model describing the overall structure of the data and the relationships between the data (Hellerstein, Stonebraker 2005). The data models vary from flat file data models, to hierarchical, network, relational and to object-oriented models. They are tightly coupled to the DBMS. While the data model is a description of the overall structure of the data and their relationships, the DBMS uses this description for storing and retrieving data.

Figure 3.1. Definitions and illustrations of databases
3.1.2 Types of Database
There are two types of databases: manual and computerized (Berg, Seymour & Goel 2013). As illustrated on the right hand side of Figure 3.1, a manual system, as its name indicates, involves manual data storing and processing. For instance, a university uses paper documents for storing information about students, teachers and its operation. A process for managing university data is considered manual as long as its data are stored by humans on non-electronic media.

A computerized database is a system involving computerized data processing. For instance, the university uses computer applications that store and manage all the data electronically. A process of managing university data is considered computerized when its data are stored by computers on electronic media. There are many types of semi-computerized and computerized databases. We group them into pre-relational, relational and post-relational databases. (Hellerstein, Stonebraker 2005)

3.1.3 Pre-relational Databases
Pre-relational databases are represented by two types: file-based systems and navigational databases. Two of the navigational databases that are dominant are hierarchical and network databases (Berg, Seymour & Goel 2013). The difference between the manual and computerized databases lies in who navigates and how. We classify the pre-relational databases as semi-computerized. In file-based systems, it is a programmer who manually navigates a database via the data structures stored in flat files. In the navigational databases, on the other hand, it is the database management system that interacts with the user/programmer and assists in a manual navigation of the data structures that are stored in the database. These data structures must be known and followed by the applications using the database. In this section, we briefly describe the pre-relational databases. Section 3.2.1 describes traditional file-based systems. Sections 3.1.3.1 and 3.1.3.2 give an account of hierarchical and network databases, respectively.
3.1.3.1 Traditional File-Based Systems

Traditional file-based systems are the first computerized systems. As illustrated in Figure 3.2, they consist of one or several applications that are developed for providing services to specific end users such as, for instance, university. Applications, in turn, manage their own data records that are stored in flat files, the files that contain records with relationships. The data structures and relationships are declared and stored in application programs in form of data records and/or variables.

The file-based systems work well as long as their number of records and relationships is small. As soon as the size and complexity of data grows, it is hard to process information in the files and maintain their relationships. Following Figure 3.2, one can see that data on one and the same employee may be stored in different files and managed by different programs. This, in turn, leads to redundancy and risk for introducing inconsistencies into the files. (Berg, Seymour & Goel 2013)

3.1.3.2 Hierarchical Databases

A hierarchical database is a data model that structures data relationships in a hierarchical way. As shown in Figure 3.3, all data originates from a single table which acts as a root node. Other tables then branch out from the root node creating a tree structure. The relationships in the hierarchical model can be thought of as relationships between parents and children. Parents can have many relationships with children, but a child can only have one parent. This rule allows for data to be systematically accessible. To get to a low-level table, you start from the root working your way down to the table that you wish to access. (Berg, Seymour & Goel 2013)

![Figure 3.3. Illustrating hierarchical and network data model](image-url)
The hierarchical model handles redundant data somewhat better than the file-based systems. Now the information is centrally stored in a single database and not widely dispersed across many different files. Redundant data may still exist since the model only allows one-to-many relationships. For example, if a table Course is a parent to the table Student and the student attends many courses, then several instances of the same student record need to be added for each course instance, thus creating redundancy. Ability to store many-to-many relationships is needed since one student may enroll on many courses, and a course may have many students. (Hellerstein, Stonebraker 2005)

3.1.3.3 Network Databases

The network database model presents the data in a graph structure with tables storing related data. As illustrated in Figure 3.3, the tables act as nodes and the relationships act as arcs. The structure does not define restrictions to its relationships meaning that one node may have many relationships with multiple nodes. Many-to-many relationships are managed by splitting them into two one-to-many relationships. As shown in Figure 3.3, the record Course/Student is created. This table stores data for a specific student and course.

The network data model was a direct solution to the problem presented in the hierarchical model. While the hierarchical model created a tree-like structure restricting the relationships among the nodes, a network data model allowed relationships to be freely added thus eliminating the hierarchy. Allowing the split of many-to-many relationships was a substantial improvement because of the extended ability to implement more complex relationships. Despite this, the network implementations were still too complex to accommodate the growing need for managing large amounts of complex data. (Hellerstein, Stonebraker 2005). They burdened programmers with substantial navigational effort.

3.1.4 Relational databases

The relational model is based on the foundations of mathematics built on the concept of relational algebra (Berg, Seymour & Goel 2013; Connolly, Begg 2010, p. 92-95; Hellerstein, Stonebraker 2005). As illustrated in Figure 3.4, data in the relational model is stored in tables and the tables consist of rows and columns. A row in a table describes a certain instance and an instance can be accessed by using identifiers known as primary keys. Tables implement relationships with themselves by storing foreign keys in the columns referring to the related tables. Information is retrieved by comparing the data value you wish to retrieve with search criteria written in a declarative programming language called Structured Query Language (SQL). (ibid.)

Just as in the network databases, the relational model provides both one-to-many and many-to-many relationships thus making the data structure less complex and difficult to understand. For example, as illustrated in Figure 3.4, many-to-many relationship is represented with the joint table Course/Student. This table stores the primary keys of two tables (Course and Student) thus splitting up the many-to-many relationship into two one-to-many relationships.

The relational model differentiates itself from both the hierarchical and network based models where users need to understand how to find the data in their complex structures. It relieves its users from the error-prone and complex navigations thanks to the concepts of tables, primary and foreign keys and SQL (Hellerstein, Stonebraker 2005). In addition to the above, relational databases provide useful tools for database administration. The database itself provides constraints, access rights, integrity, data validation and other useful mechanisms. This minimizes the gap between database administration and database usage. (Connolly, Begg 2010, p. 95-105)
3.1.5 Post-relational Databases

At the end of the twentieth and at the beginning of the twenty first century, new types of databases have started emerging. These were object-oriented and NoSQL databases. They arose due to the problems encountered in relational databases. In this section, we give an account of these new databases. Section 3.3.1 describes object-oriented databases. Section 3.3.2 provides an overview of NoSQL databases. (Berg, Seymour & Goel 2013)
3.1.5.1 Object-Oriented Databases

Object Oriented databases (OODs) store information in form of objects as used in object oriented programming languages (Berg, Seymour & Goel 2013). OOD’s aim is to incorporate the object oriented principles such as encapsulation, polymorphism and inheritance. In addition, OODs implement database concepts such as system integrity by atomicity and consistency. (Hellerstein, Stonebraker 2005)

OODs have become popular thanks to the absence of impedance mismatch (Berg, Seymour & Goel 2013). This means that applications written in object oriented languages are directly mapped to the database model. Therefore, it is easy for developers to understand the structure of both the application and the database. However, this also presents language dependence. An OOD is typically written for a certain programming language; hence, applications written in other languages cannot directly access the database. Furthermore the object oriented approach may cause problems when a need arises to analyze data that do not correspond to the object oriented structure. (Berg, Seymour & Goel 2013)

Figure 3.6. Illustrating NoSQL databases
3.1.5.2 NoSQL Databases

In protest against the constraints imposed by the relational databases, many new different database technologies have been suggested. On one conference in San Francisco, they were grouped as NoSQL databases (Sadalage, Fowler, 2013). These databases, however, do not have any common definition. Neither do they rely on any common technology nor do they have any authority responsible for them. Generally, NoSQL databases are grouped into four different categories. These are (1) **key-value stores**, (2) **column-family stores**, (3) **document-data stores** and (4) **graph data models** (ibid.).

The first three types of NoSQL databases can further be classified as aggregate data models. An aggregate data model is a database model designed to handle large amounts of data and to aggregate the data in an efficient way. This means that their data models group together collections of related objects and treat them as units. (Sadalage, Fowler 2013; Shah, Wei & Kolovos 2014)

The fourth data model, the graph data model, is typically used for storing relationships. For example, new web companies need to store a selection of links and related data. These data do not have any underlying data models. They are only linked together, sometimes by chance. For example, Google uses a graph model to store related links and Facebook uses it to store related posts (Cayley, 2015; Facebook, 2015). In this thesis, graph data models will be excluded from this study. Therefore, when referring to NoSQL databases in this thesis, we are referring to a database that consists of an aggregate data model. Below, we describe the three NoSQL databases fitting into this framework. Our descriptions are supported by Figure 3.6.

**Key-value store**

Key-value store is a data model with a structure of a map, such as for instance, a hash map. As illustrated in the upper left hand side of Figure 3.6, the key is used as an identifier for searches and the value store includes the aggregate. There is no other way to look up a certain aggregate (value in key-value store parlance) than through the key. The value is opaque to the database meaning that the database has no clear definition of what data is inside the value column. This is a simple structure but efficient for searches that can be based on a key without having to know the structure inside the value. In the industry, key-value databases are used for managing less complex data (Li, Manoharan 2013; Sadalage, Fowler 2013; Shah, Wei & Kolovos 2014).

If we were to store information about courses in a key-value data store, the information about a course ID is stored under one key and the information about the course as well as teachers and students that attend the course is stored in the value store. The teachers and students are stored as a list.

**Document-data model**

The document data model is based on the same foundation as the key-value store. Document-data model has also a key that users may use to look up certain data. However, in contrast to the key value stores, one does not need have access to the key in order to retrieve data. Together with the other data, the key is stored and retrieved in the document value store. Because the document value store has a clear structure, users can submit queries to the database based on certain fields stored in it. (Li, Manoharan 2013; Sadalage, Fowler 2013; Shah, Wei & Kolovos 2014).

The upper part of the right hand side of Figure 3.6 illustrates a document-data model when structuring a document data model with regards to the example of teachers, courses and students. The data structure looks similar to the key-value store. The difference is that the search is not reliant on the key. It can be based on any value as shown in the figure. Furthermore, the document data model allows
documents to be updated dynamically so if students or teachers were not to be part of the course any longer, they would then be deleted from the document with a simple query.

**Column-family store**

Column-family data stores includes different types of structures. They are all based on foundations of keys. However, the value itself is stored in a different way compared to the previous data models. The column-family data model considers the fact that certain data are often accessed together. Such data forms a column-family. When accessing a certain column-family, the other data is ignored. This minimizes the space in which the search is conducted, and thereby, enables faster retrievals. Even if a row has over one hundred columns, the retrievals only target a specific column-family, and ignore all the other data. (Li, Manoharan 2013; Sadalage, Fowler 2013; Shah, Wei & Kolovos 2014)

If we structure data using our previous example with students, teachers and courses, there are a number of different ways in which data can be organized in a column-family store. The example shown on the bottom part of Figure 3.6 uses the notion of column families. The information about the course is stored together in one column family called `course_info`, the information about the students is stored in the second column family called `student_info`, and the information about the teacher is stored in the third column family called `teacher_info`. This allows one column family to be accessed while the other column families get ignored.

### 3.2 Database Selection Process Models

Despite a comprehensive literature study and extensive internet exploration, we could not find any model dealing with database selection process. Hence, we cannot present any closely related work herein. We have only found publications dealing with related process models. Those concerned processes such as software system reengineering, migration, replacement, modernization, benchmarking, mining, application selection and the like. All of them mainly focused on software applications and very few of them touched upon the subject of databases. The majority of them, however, included databases as their integral part.

Many organizations have large portfolios of legacy systems that they either continue to evolve and maintain or that they have to replace with other systems. For this reason, much research has been done in the domains of modernization, replacement and migration of software systems. Due to space restrictions and the fact that the research reported herein is only indirectly related to database selection, we cannot report in detail on them all. Instead, we will list various issues that have to be considered in those processes. For simple referencing, we also group all those processes under the name of **endeavor processes**.

Due to the fact that our SeDB process model ends up after a database gets selected, the issues of the endeavor processes presented herein are mainly focused on their initial phases. They neither cover installation nor changes to the surrounding software, hardware and business environments. They mainly encompass three phases: (1) **Pre-Decision** during with one studies the software system to be modernized, replaced or migrated, (2) **Decision** phase during which one decides on whether to start the endeavor process, and (3) **Candidate Benchmarking and Final Database Selection** during which the candidate databases are benchmarked and the most appropriate database gets selected. The three phases are presented in Sections 3.2.1-3.2.3, respectively.
3.2.1 Pre-Decision Phase
The first phase of the endeavor process deals with identifying a need for acquiring a database and with analyzing the situation within the company to assure that the fulfillment of the need is justified. The following steps of the Pre-Decision phase have been identified within the literature studied:

**Identify a need for starting a new endeavor**
The idea of modernizing, replacing or migrating a software system should be based on a need for starting the endeavor (Bernonville, Kolski & Beuscart-Zephir 2007; SEMAT 2013). A need is often experienced due to some problems such as poor software system quality or insufficient support of the software or hardware platforms.

**Evaluate the use of the current system**
Software system may be used by a large or a small number of users. This may be an important criterion for evaluating its business value. To make a fair evaluation, however, a system in use must be examined from a longer time perspective. Certain applications may be intensively used within only a limited period of time and still be of key value to the organization. A representative example of it is a student registration application that is only being used at the beginning of each semester. Still, it provides a key functionality to a university management system (Somerville 2011, p. 252).

**Evaluate the system understanding**
Developers are often bothered with understanding legacy systems* (Wu, Sahraoui & Vatehev 2005). Reasons are many. Often, it is a combination of lack of or inconsistent documentation. This does not only obstruct the system evolution, but also makes the endeavor process very complex and error prone. Systems that are difficult to understand should definitely be reworked in some way. One should keep in mind the challenges of extracting and preserving the business logic from the documentation of such systems (ibid.).

**Assess the business value of the software system**
All software systems must be evaluated from the business value perspective. They are the result of large investments and they store most of the business knowledge and expertise. Identifying the business knowledge is the fundamental starting point in all types of endeavors (Lucia, Di Lucca 1997). In cases when companies have difficulties to assess business value, they have to retrieve the embedded business knowledge from the legacy systems (Perez-Castillo, 2012). If the business value is low, then it is meaningless to keep the systems in operation. Such systems are costly, and therefore, they should be removed from the business operation (Somerville, 2011 p. 252).

**Determine software system quality**
Determining software quality is of importance in relation to the business value when it comes to making decision on how to proceed. As Sommerville (2011) suggests, low quality systems having low business value should be removed from the operation whereas high quality systems with high business value should be kept in operation. Low quality systems with high business value may be replaced and high quality systems with low business value may or may not be kept in operation. To determine system quality, various quality attributes have been defined which the companies may use for determining their system quality (Berander et al. 2005). The companies have to determine on their own which of those quality attributes are pivotal for determining the system quality. Examples of them are provided in Figure A.1 in Appendix A. Their importance varies from case to case. However, the ones that are considered of great importance in the endeavor context are modularity, complexity, documentation, changeability, stability and testability (Zou, Kontogiannis 2002).
Check the supplier stability
Supplier stability is one of the most important concerns when deciding to migrate, replace or modernize a system (Somerville 2011). This concerns the suppliers of the new and old systems. Here, one should consider whether the supplier is going to exist in a predetermined period of time in the future and how financially stable the supplier is.

Evaluate the hardware and software environment
The hardware and software environment must always be considered. It is not enough that the system is of high quality. Its environment may suffer from many deficiencies such as age, failure rate, performance, maintenance cost, interoperability, support cost and the like. To assure that all aspects of the hardware and software environment are paid heed to, checklists including all those items have been suggested (Collier et al. 1999; Somerville 2011).

Evaluate fulfillment of system requirements
Current systems must be evaluated from the perspective of how they fulfill current system requirements. The same concerns the target systems. All types of requirements should be evaluated such as functional and non-functional. Of importance is it to consider the current and target architectural requirements due to the fact that they devour most of the endeavor resources (Breivold, Larsson & Land 2008).

Determine changes to be made
Various endeavors (modernization, replacement or migration) require various levels of changes. Those must be identified and, if possible, quantified. Example of changes may range from rewriting some applications in a more modern programming language to migrating to the cloud.

Assess impact of the endeavor
All changes have impact on software, hardware, organization and business environment (Hughes et al. 1997). The impact may concern various things, from a need for changing the surrounding applications to the immense effort to educate and train all the people involved in changing and/or using the system to eventually changing business operation. The impact for all levels of changes must be identified and clearly quantified.

Assess effort of the endeavor
Using the information on the changes and impact of the endeavor, one should assess the overall effort of the endeavor. The effort depends on the level of changes and the level of impact (Sommerville, 2011)

Conduct feasibility study
Having all the information available, one should conduct feasibility study in order to objectively identify strengths and weaknesses of the endeavor. Here, one mainly focuses on the cost of the endeavor, the value to be gained and the chances of succeeding with the endeavor. In this activity, one should assess the ease of the endeavor (Balali et al. 2010), identify conflicts preventing the endeavor (Johnson, 2002) and investigate the alternatives to the endeavor (Bjornestedt, 2009).

There may be many conflicts preventing the implementation of the endeavor. An example of such a conflict may be a human conflict such as inertia towards change (Stelzer, Mellis 1999). Another example may be a technological conflict such as choice of a specific application due to the requirement stating that the existing software and/or hardware platform must be kept (Ibid.).
Many organizations decide to proceed with the selected endeavor type without putting enough resources into investigating alternatives. For instance, while improving system performance, instead of migrating a system to a new software and hardware platform, they might consider restructuring the code, and thereby, probably improve its performance (Bjornestedt, 2009).

**Determine appropriate strategy for how to proceed**

After having assessed all the above, the organizations should determine a strategy on the next steps. The strategy may implement the endeavor either in a big bang or in an iterative manner. (Somerville 2011)

**Ensure communication across organization on the status of the modernization, replacement or migration process**

All people involved in the process must be informed about the endeavor on a regular basis. This does not only concern communication between business management and engineering groups but also the actual users of the system and customers (Breivold, Larsson & Land 2008).

### 3.2.2 Decision Phase

During the *Decision* phase, the organization makes a decision on whether to proceed with the endeavor or not (Somerville 2011). An important question to be answered is whether the endeavor is necessary (Bjornestedt, 2009). The decision should be based on the answers together with all the feedback of the *Pre-Decision* phase. It may either result in “Continue with the endeavor”, “Rework the results of the Pre-Decision Phase” or “Do not continue with the endeavor”.

### 3.2.3 Candidate Selection

After having decided to invest in the endeavor, the candidate applications get selected. Those are then benchmarked with respect to the attributes that are deemed to be relevant for their selection (Beckman 1990; Märgner, Karcher & Pawlowski 1997). The attributes typically concern quality such as performance, scalability, time, cost, supplier support and the like. The benchmarking may be either purely human or automatic. By human, it is meant that humans evaluate the benchmarking output whereas in the automatic benchmarking, it is the computers that do it (Märgner, Karcher & Pawlowski 1997).

### 3.3. Feedback to Our Work

Both the theory on databases and endeavor processes presented herein constitute an important feedback to our work. In this section, we describe how they supported our thesis. Section 3.3.1 focuses on the feedback from the database technology perspective, whereas Section 3.3.2 does it from the software process model perspective.

#### 3.3.1. The Database Technology Perspective

In today’s database market and research, all of the presented databases are still in use. However, some databases are more popular than the other. Many of the databases have developed niche markets while others are used in various settings and organizations.

File based systems are still used today because of the increasing need for storing unstructured data in various contexts. For instance, the file system in Windows is file based (Windows files, 2015). Hierarchical databases are still used in applications that demand high performance and availability. For example, banking systems and telecommunications still use IMS databases (Berg, Seymour & Goel 2013). Network databases are used in systems that demand complex relationships between data. Just as hierarchical databases, they are used in banking applications and telecommunication systems
New upgrades of both the hierarchical and network databases are still available and they are highly popular (Hellerstein, Stonebraker 2005).

Relational databases are the most popular. They are used in almost all types of organizations with different purposes. Typical examples are various administrative systems. Their early challengers, the object oriented databases, are only used in some niche markets today. Examples of those markets are artificial intelligence and computer-aided design and manufacturing (Hellerstein, Stonebraker 2005).

Recently, NoSQL databases are becoming increasingly popular because of the ease of storing large amounts of unstructured data. This is especially suited for the new era of web 2.0. Many of the big web giants such as Amazon, Google and Facebook have been pioneers and developed the NoSQL databases that are popular today. Some examples of them are Cassandra, HBase and Google Big Table. (Li, Manoharan 2013, Sadalage, Fowler 2013)

3.3.2. The Process Model Perspective

The SeDB process model that is suggested in this thesis considers all types of databases. It also considers the majority of the steps of the endeavor processes as presented in Section 3.2 and adapts them to the database selection domain. Although there are many suggestions for modernization, replacement and migration process models, they all differ in their level of detail and inherent activities. When gathering their process constituents, we found them scattered in more than forty publications. By collecting them, we acquired compiled knowledge of what needed to be done in endeavors related to our work.

The related work made us aware of various aspects of a database selection process models that need to be considered. Those concern purely business aspects such as, for instance, identification of a business need and value that need to be considered upfront before taking any other steps. Other aspects to be paid heed to are of engineering character, such as evaluation of the system quality undergoing the endeavor, evaluation of the system environment, identification of the requirements that need to be fulfilled by the system, and the like. Those aspects are often intertwined with technological, economical and human aspects such as investigations of the new systems and stability of their suppliers, assessment of the impact of the endeavor, feasibility studies and choice of strategies for implementing the endeavor. Of great importance are training and information dissemination aspects across the whole organization.
Chapter 4. Evaluation Model

This chapter presents the evaluation model that has been used to evaluate the SeDB process model. Section 5.1 gives an overview of the evaluation model and Section 5.2 describes it in detail. Finally, Section 5.3 explains which evaluation criteria were used in which evaluation process phase.

4.1 Overview of the Evaluation Model

There are two main ways of evaluating newly suggested process models: (1) by benchmarking them against industrial process models or other published models, or (2) by interviewing experts within the process domain. Unfortunately, we neither had the luxury of evaluating our process model against the industrial processes nor against any published models. As mentioned earlier, there are no published models specifically designed for selecting databases. Therefore, we had to evaluate them using interviews with experts. Only one minor part of the model was evaluated with action research.

It is difficult to evaluate designed process models. According to Sadera, Rosemann & Gable (2002), there are no clear criteria for evaluating process models and their success. Success is not well defined and it does not always apply across all process contexts (Larsen and Myers, 1998). For example, the success of the execution of a process model in one context may not result in success in another process execution context. This is because the executed process instances may strongly vary with respect to the choice of their inherent activities and the order among them.

The only concrete output of the design phase is the process model. At that time, the process model is not yet implemented in an organization. To implement it in the real world may take time and evaluating the process model after its implementation may be too risky. Many resources may be put into implementing the process model across an organization only to realize that the model was not good enough. For this reason, all process models should be evaluated before getting implemented in some way.

The evaluation of process models does not provide any tangible direct output in form of understanding of their increased efficiency, effectiveness, user satisfaction and other forms of benefit. The evaluation instead provides interpretations of the models and the fulfillment of their objectives (Sadera, Rosemann & Gable, 2002).

![Figure 4.1. Modified phases of process model evaluation as suggested by Sadera, Rosemann & Gable (2002)]
To achieve an optimal interpretation of process models, one should evaluate them from both the modeler and user perspectives. The two perspectives as suggested by Sedera, Rosemann & Gable (2002) are shown in Figure 4.1. First, the process model should be evaluated by the modeler himself/herself. Afterwards, the process model should be evaluated by the users. The user evaluation consists of two consecutive steps: (1) evaluation before implementation and (2) evaluation in use.

The SeDB process model was evaluated from the “modeler” perspective and from the “evaluation before implementation” perspective. In both cases, we evaluated “the extent to which the modelers and users believed that the process model met the fulfillment of the objectives underlying the modelled process” (Doll & Torkzadeh, 1988).

Our evaluation model focused on five criteria. These are (1) interviewee credibility, (2) semantic correctness, (3) syntactic correctness, (4) usefulness, and (5) process flexibility. Those are listed on the left hand side of Table 5.2. Each criterion was evaluated with a set of interview questions. The questions and their relationships to the evaluation criteria are presented on the right hand side of Table 5.2.

4.2 Evaluation Criteria
This section explains each of the five evaluation criteria, motivates them and relates them to the interview questions. All this is done in Sections 4.2.1-4.2.6, respectively.

4.2.1 Interviewee Credibility
Credibility is the quality of being trustworthy and believable. When interviewing people about a certain subject, it is important to ensure that the right people have been chosen, people who are trustworthy when it comes to the subject at hand. This thesis aims at presenting a process model for selecting a database. Hence, interviewees need to meet certain criteria for being believable within the domain of databases and processes. At least on one occasion, the interviewees must have participated in the process of selecting a database or similar processes. For assuring that the right people were interviewed and that they had solid enough experience, the following questions were asked:

- IC 1: What is your profession?
- IC 2: What roles have you had?
- IC 3: For how long have you had those roles?
- IC 4: Have you ever participated in any database selection project?

4.2.2 Semantic Correctness
The word semantic relates to a meaning of some words, phrases, objects or symbols. The semantic correctness, on the other hand, relates to the accuracy of their explanations. If, for instance, an object is semantically correct, then it makes sense to its perceiver.

In this evaluation model, semantic correctness refers to the accuracy of the description and explanation of the SeDB process model. Semantic correctness ensures that the model is understandable for real world projects. If the process model is semantically correct, then the process executor should be able to understand the relevancy and applicability of its steps. The semantic correctness of the SeDB process model was evaluated with the following questions:

- SeC 1: Are the process steps relevant to be part of the process model?
- SeC 2: Are the process steps applicable?
## Table 4.2. Evaluation criteria together with the related interview questions

<table>
<thead>
<tr>
<th>Interviewee Credibility</th>
<th>IC 1: What is your profession?</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IC 2: What roles have you had?</td>
</tr>
<tr>
<td></td>
<td>IC 3: For how long have you had those roles?</td>
</tr>
<tr>
<td></td>
<td>IC 4: Have you ever participated in any database selection project?</td>
</tr>
<tr>
<td>Semantic Correctness</td>
<td>SeC 1: Are the process steps relevant to be part of the process model?</td>
</tr>
<tr>
<td></td>
<td>SeC 2: Are the process steps applicable?</td>
</tr>
<tr>
<td>Syntactic Correctness</td>
<td>SyC 1: Are there any redundant or unnecessary steps in the proposed model?</td>
</tr>
<tr>
<td></td>
<td>SyC 2: Are there any important steps related to selecting a database that are not addressed in the process model?</td>
</tr>
<tr>
<td></td>
<td>SyC 3: What do you think about the proposed sequence of steps in the process model?</td>
</tr>
<tr>
<td>Process Flexibility</td>
<td>F 1: With minor adaptations, is it possible to use the SeDB process model in various database selection projects?</td>
</tr>
<tr>
<td>Usefulness</td>
<td>U 1: Does the SeDB process model guide you in selecting a database?</td>
</tr>
<tr>
<td></td>
<td>U 2: Would you consider using SeDB process model for selecting a database?</td>
</tr>
<tr>
<td>Experience</td>
<td>E 1: Do you see any problems related to the individual steps of the process model?</td>
</tr>
<tr>
<td></td>
<td>E 2: Do you see any problems related to the whole process model?</td>
</tr>
<tr>
<td></td>
<td>E 3: Can you describe some of the major challenges of selecting a database?</td>
</tr>
<tr>
<td></td>
<td>E 4: From your experience, what area do you think is the most challenging in a database selection process?</td>
</tr>
<tr>
<td></td>
<td>E 5: Can you briefly tell us about the lessons learned from any database selection project you or your company have initiated/accomplished?</td>
</tr>
<tr>
<td></td>
<td>E 6: Do you have any other opinion to share about the process model?</td>
</tr>
<tr>
<td></td>
<td>E 7: Do you have any suggestion you would like to share to improve the process model?</td>
</tr>
</tbody>
</table>

The evaluation of the semantic correctness was done to ensure that the process steps and concepts presented in the model had accurate meaning. Question SeC 1 inquired whether the steps had a significant and demonstrable bearing on a database selection process. Question SeC 2 explored whether the steps were suitable for the process.

### 4.2.3 Syntactic Correctness

The word *syntactic* relates to the meaning of some words, phrases, objects or symbols. In linguistics, syntactic correctness refers to the combination of morphemes* that, when put together, form another word or meaning. For example, the word *unbreakable* consists of three separate words. These are (1) “un”, (2) “break”, (3) “able”. When the three words are put together, they form a different word. Syntactic correctness refers to the order in which the words are combined. If the words are put together in a different way, they would not be syntactically correct.

In our model, syntactic correctness refers to the logical steps of the process model. The steps are not treated as single units. Instead, they are viewed holistically to ensure that they are coherent. Syntactic correctness was evaluated with the following questions:
• **SyC 1**: Are there any redundant or unnecessary steps in the proposed model?
• **SyC 2**: Are there any important steps related to selecting a database that are not addressed in the process model?
• **SyC 3**: What do you think about the proposed sequence of steps in the process model?

The goal of questions **SyC 1** and **SyC 2** was to see if the SeDB process model either missed any important steps or if the steps were unnecessary or redundant. Then, question **SyC 3** was asked to elicit the opinion about whether the sequence of the steps was appropriate.

### 4.2.4 Process flexibility

Flexibility is the capability of adapting and adjusting easily to the external changes. In this study, we focused on the flexibility of the SeDB process model, that is, on its ability to be adaptable, adjustable and reused in various process instances. This criterion is important to ensure general usability* of the SeDB process model and to confirm its transferability to various contexts. The following question was asked:

• **F 1**: With minor adaptations, is it possible to use the SeDB process model in various database selection projects?

### 4.2.5 Usefulness

Usefulness* refers to something being applicable and of practical use. The process model should serve some purpose that makes it helpful when put into practice. In this thesis, we studied the usefulness of the SeDB process model with the following questions:

• **U 1**: Does the SeDB process model guide you in selecting a database?
• **U 2**: Would you consider using SeDB process model for selecting a database?

### 4.2.6 Experience

Experience is the skill or knowledge gained by doing something. Experience is very important. Individuals having solid process experience have encountered many problems and challenges throughout the years. They have also learned many lessons aiding them in tackling those problems and challenges.

The aim of the experience criterion was to elicit problems, challenges and lessons that the interviewees encountered and/or learned and to relate them to the SeDB process model. The criterion was evaluated with the answers to the following questions:

• **E 1**: Do you see any problems related to the individual steps of the process model?
• **E 2**: Do you see any problems related to the whole process model?
• **E 3**: Can you describe some of the major challenges of selecting a database?
• **E 4**: From your experience, what area do you think is the most challenging in a database selection process?
• **E 5**: Can you briefly tell us about the lessons learned from any database selection project you or your company have initiated/accomplished?
• **E 6**: Do you have any other opinion to share about the process model?
• **E 7**: Do you have any suggestion you would like to share to improve the process model?

We expected that the answers to these questions would provide us with feedback that might be useful to detect problems with the SeDB process model or feedback that might additionally strengthen the choice of its inherent activities.
4.3 Mapping Evaluation Criteria on the Evaluation Process
We conducted two types of the evaluation of the SeDB process model. Using Figure 2.2, these were (1) evaluation of the description of the whole SeDB model using the interviews in the Evaluate SeDB Round 1, Evaluate SeDB Round 2 and Evaluate SeDB Round 3 phases and (2) active evaluation of the latter part of the SeDB model in the Benchmark Databases phase. Except for the credibility, process flexibility and experience criteria, we used all the evaluation criteria in both evaluation types. The difference in the choice of the criteria used lied in who evaluated the model. The interviewees did it in the first interview-based type whereas we did it in the second action research based type. When evaluating the SeDB model via action research, we considered all the interview questions for each of the three criteria (semantic, syntactic and usefulness) and tried to answer them while conducting action research.
Chapter 5. Preliminary SeDB process model and its First Round Evaluation

This chapter presents the preliminary SeDB process model and the results of its first round evaluation. Section 5.1 describes the preliminary SeDB whereas Section 5.2 presents the results of the Round 1 interview with the academic interviewee. Finally, Section 5.3 presents our evaluation of the interview results.

5.1 Overview of the preliminary SeDB process model

The Select DataBase (SeDB) process model is a suggestion for a concise and simplified description of the steps that need to be considered while selecting a database. As shown in Figure 5.1, it consists of seventeen activities and twenty-three sub-activities. In this chapter, we only give a brief account of those activities and motivate their order. Due to space restrictions, we moved their detailed descriptions to Appendix C.

The very first step of the SeDB process model deals with the identification of a need for a new database (Activity 1). Companies have always restricted budgets and resources and cannot afford to acquire a new database on an impulse purchase basis. Hence, all acquisitions must be strongly supported by a need. It is not enough, however, to only have a need. One should also evaluate whether the new database contributes to business. For this reason, companies should identify a business opportunity to be gained thanks to the acquisition of a new database (Activity 2). Business opportunity may only be determined by evaluating a gap between current and target business models (Activity 3, Activity 4 and Activity 5). Finally, companies should find out how the acquisition of a new database impacts the organization (Activity 6).

To get a thorough understanding of how the business will be affected, companies should estimate the cost of acquiring a new database (Activity 7). They have to objectively evaluate whether the potential success is credible and feasible (Activity 8), and they have to identify and assess all the risks that may be encountered during the acquisition endeavor (Activity 9). Finally, they should assign budget for the endeavor (Activity 10) and make a decision on whether to proceed with the database selection (Activity 11). While making the decision, they should not forget to inform the employees about it (Activity 12).

After the previous steps have been completed, the organizations should now move forward with the database selection. Here, they should elicit requirements guiding the selection of a database (Activity 13) and select candidate databases (Activity 14). Those candidate databases should then have to be evaluated against the requirements as stated in Activity 13. Using the requirements, the organizations should then narrow down the number of databases (Activity 15) which they should then test in an appropriate testing environment (Activity 16). The testing results should help the organizations choose the database that fits their organizational needs (Activity 17).

5.2 Round 1 Evaluation of the SeDB process model

In this section, we present the results of the first round evaluation with our academic interviewee. Before being interviewed, the academic interviewee received our description of the state of art within research that is presented in Section 3.2, the preliminary SeDB process model as described in Section 5.1, and the questionnaire as described in Chapter 4. While presenting the results of the interview and their analysis, we follow the order of the evaluation criteria and their respective questions as defined in
Chapter 4. To remind our reader, the order is the following: (1) interviewee credibility, (2) semantic correctness, (3) syntactic correctness, (4) process flexibility, (5) usefulness, and (6) experience.

5.2.1. The Academic Interviewee Credibility

The academic interviewee was of the opinion that she had enough competence to answer all our questions. She is associate professor in software engineering and senior lecturer at one Swedish University. She has had this profession for more than two decades. The scope of her research covers the whole software lifecycle ranging from the conception of an idea for developing a software system until software system retirement. A subset of subjects she has dealt with are development, evolution and maintenance of software systems, front-end and back-end software support processes, emergency problem management, pre-delivery maintenance, software handover, retirement, release management, SLA management, competence development, risk management, documentation, testing and quality assurance, agile methods, software process improvement, reengineering, process flexibility and tailoring, globalization and various adaptations to new technologies and methodologies.

Our interviewee has not directly conducted research within database selection process. However, she has created models for several unexplored processes. In the world, she is the only author or one of the very few authors of software retirement, software handover, pre-delivery maintenance, SLA management and the like. In addition, she has experience in exploring unexplored software engineering processes and she has been involved in doing research in strongly related subjects, such as reengineering, handover, migration to cloud, and other subjects. Finally, our academic interviewee has co-created ESSENCE, an international OMG software engineering standard that constitutes a theoretical basis for all kinds of software engineering methods (SEMAT, 2013).
5.2.2 Semantic Correctness

The academic interviewee was of the opinion that the process steps were relevant to be part of the database selection process. According to her, all of the steps are applicable and necessary. She was glad to see that we included many business related activities such as identification of a need and opportunity and evaluations of the current and target business models. According to her, many endeavors often forget these very important activities. Need and opportunity should always be the driving forces of the database selection process. Unfortunately, there are still some companies that forget it and decide to choose a new database or software application without giving a thought to the need. The selection of a new database may be often driven by some strongly motivated and opinionated individuals and little heed is being paid to the evaluations of current and target business models.

5.2.3 Syntactic Correctness

The academic interviewee provided us with much feedback on the syntactic correctness of the SeDB process model. She was of the opinion that the SeDB model did not include any redundant or unnecessary steps. However, it lacked some very important activities. These were planning, identification of problems and their consequences, identification of the stakeholders involved, determination of a type of a database, and accountability.

The planning step is an important and integral part of any process irrespective of whether it is one-off or continuous. Here, she quoted a famous proverb “failing to plan is planning to fail”, as widely attributed to Alan Lakein, the writer of several books on management (Lakein, ud). Planning helps identify and map out a sequence of steps with the purpose of achieving some goal. Proper planning reduces much time and effort and helps organizations follow their progress. If the organizations do not plan, then they may arrive at an unstructured and chaotic process in which they either take many unnecessary and unfocused steps or they may spend too much effort on determining what to do next. Planning helps organizations know where they are, which is essential for succeeding with any engineering or non-engineering endeavor.

Regarding the identification of problems and their consequences, our interviewee stated that these steps were strongly related to a need. However, there may not always be one to one relationship between those two. All kinds of businesses encounter problems, and therefore, the identification of a need or problems should serve as a starting point in all kinds of software engineering endeavors. A problem should be well-articulated and its consequences should be well understood. According to the academic interviewee, problem and need often go hand in hand. Usually, a problem may generate a need, however, a need may not always be generated by a problem. For instance, a small company using a simple Excel program may not experience any problems with it. However, they may have a personal need to go over to Access, a database offered by Microsoft (Microsoft Access, 2015).

All kinds of software engineering endeavors are human intensive. According to our academic interviewee, the SeDB model failed to recognize the involvement of various stakeholders. Stakeholders are people, groups, or organizations who affect or are affected by the database selection process. It is they who experience problems and need, provide opportunity, state requirements, fund the database and database selection process, conduct database selection process, make important decisions, and the like. They are involved throughout the database selection process and ensure that the right database gets selected. Hence, they are critical to the success of the database selection process. Their feedback helps shape the process and ensure its quality.
Regarding the accountability, our academic interviewee wished it to be included in the model. However, as she pointed out, there are many strongly varying stakeholders and identifying them all may be too much for our master thesis. As a compromise, she suggested that we used the SEMAT’s idea of grouping the stakeholders into the groups that were best suited for our process (SEMAT, 2013). The groups she suggested were (1) stakeholders that fund the database selection process, (2) stakeholders that use the database, (3) stakeholders that conduct the database selection process, and (4) stakeholders that represent various parts of the organization or collaborating organizations.

It is important that the organizations first study which type or types of a database suit(s) their needs. In our interviewee’s parlance, the organizations cannot afford to “jump” from one database type to another. Many banks, for instance, still use hierarchical databases and intend to do so in the near future. Going over to relational database may imply substantial impact on the organization’s echo system*. They may have to rewrite all the applications, educate and train almost all the employees, change their business processes and the like. This may be too costly. Therefore, before selecting candidate databases, the organizations should agree upon which type of a database they wish to acquire.

Regarding the order of activities, the academic interviewee had a divided opinion. According to her, the logical order of the steps is correct, however, in reality, process steps are always context dependent. The SeDB process model, as it looks now, is a strongly prescribing model. It does not include any information on how companies may deviate from the order of its steps. Neither does it provide any guidelines why and how they may deviate from the process model so that the database selection endeavor does not get jeopardized.

5.2.4. Process flexibility
The academic interviewee was of the opinion that the SeDB process model was not flexible at all. According to her, we now live in an agile world to which we should adapt. There is much research going on how to define process models and make them flexible and easily adaptable to the continuously changing environment. The SeDB model, as it looks now, does not allow the change of the order of the activities and it seems to include a traditional one-off decision making. The former type of one-off decision making is now being replaced with continuous and iterative decision making based on what is known at the moment of making a decision.

After getting this feedback, we asked an additional question: “so, how should we organize the SeDB process model?”. Here, the academic interviewee provided us with much feedback. She suggested that we defined (1) activity types, (2) activity spaces, (3) blueprint – an exemplary process model, and (4) guidelines for when and how companies may deviate from the exemplary model.

Regarding the activity types, the academic interviewee suggested that we grouped the activities according to their function. Using them, she then suggested that we used the concept of activity spaces as suggested by SEMAT in ESSENCE (SEMAT, 2013). She drew a picture of how it worked. As shown in Figure 5.2, the activity spaces are just empty process boxes to be filled in with activities. These activity spaces may consist of sub-activity spaces and so on. They may correspond to a whole process, process phase or activity. The dashed lines in Figure 5.2 show how the activity types may be placed within the activity spaces. The placement of the activity within the activity space corresponds to the planning or execution of the activity. The activity spaces may be filled in with any activities as long as it is reasonable to have them there. Therefore, the planned or executed process models may differ from the suggested blueprint of the SeDB model. They may still, however, be correct as far as they suite a particular context.
Companies should have some kind of an exemplary process model, a blueprint indicating which activity type primarily belongs to which activity space. However, the organizations should feel free to place any activity within any activity space as far as they find it motivated. By doing so, the organizations may choose their own order that is adapted to their own context at hand. They should be however aware that if they do not choose any activity in the blueprint, they may encounter certain risks. For instance, by not studying business opportunity, they may choose a new database, put much effort and resources into acquiring and implementing it, and still, not gain any business value out of it. For this reason, the organizations should have a blueprint of an exemplary process model with explanations why its inherent activities are necessary and what risks companies may encounter if they neglect a particular activity. As illustrated with solid lines in Figure 5.2, the blueprint consists of a set of activities collected from the activity groups.

5.2.5 Usefulness
The academic interviewee found the activities in the SeDB process model useful and would consider using them in education. She would however adapt the model to the new agile trend and make them even more useful. Despite this, she believed that the model as it was now was a good basis for achieving a common understanding of what activities must be included in such a process. According to her, the SeDB process model provides a concrete roadmap against which companies may define, examine, control and improve their new or current processes. In this way, the model may guide the course of action. By suggesting a logical order, it imposes some structure and consistency on the process. Being used as a blueprint, it may help organizations find inconsistencies, redundancies, and omissions in their own processes.

To adapt the SeDB process model to the new agile trend, the academic interviewee suggested that some activities be conducted on a continuous basis and in parallel with other activities. This mainly concerns activities such as risk management, information dissemination, decision making, planning and educating and training. To make the model more useful, she suggested that the SeDB process model be put in the context of its echo system and some major software engineering endeavour. According to her, no database selection process lives an isolated life. It always supports some organization and it almost always replaces some other database.
5.2.6 Experience of the Interviewees
When being asked about the problems related to the individual steps of the SeDB process model and problems related to the whole model, the academic interviewee stated that she already pointed them out. Regarding the major challenges, experiences and lessons learned, she claimed that the major problem was process knowledge. Many practitioners do not understand the vast scope of the database selection process and they do not possess solid knowledge on how to design software engineering processes. To add zest to it, there are no database selection process models on which the organizations might base their work. Hence, the major challenge is to define such a process model.

Usually, while designing process models, the organizations learn by doing it. In the context of such a vast process, this may be a very demanding and error-prone endeavor. To define a good database selection process model may take years during which the organizations learn many useful lessons. Often, however, the lessons learned do not get documented, and thereby, they get easily forgotten. Hence, the most important lesson learned as experienced by our academic interviewee is to document all the lessons learned from one endeavor and assure that they are paid heed to in the next-coming endeavors.

5.3 Analysis of the Round 1 Interview
Although the academic interviewee has not participated in any database selection processes, we deem her knowledgeable and credible for participating in the evaluation of the SeDB process model. She has thorough experience in process models, methods and similar research areas.

The feedback we received from the academic interviewee was very useful, however, immensely comprehensive. Therefore, we had to contact the academic interviewee several times to assure that we understood her feedback. We also studied ESSENCE with the purpose of learning about activity spaces (SEMAT, 2013).

Regarding the semantic correctness of the preliminary SeDB model, we understand that it was right. All the activities were applicable and necessary. The same applies to the usefulness of the SeDB activities. The SeDB model, however, did not include many necessary activities and it suffered from flexibility problems. The missing activities were planning, the identification of problems and their consequences, identification of the stakeholders involved, determination of a type of a database, and accountability. We agree with the academic interviewee and we plan to implement those activities in the next improved version of the SeDB process model.

As mentioned by the academic interviewee, the structure of the SeDB process model did not allow flexibility that was strongly sought after in today’s industrial world. We agree that the model felt too rigid and too prescriptive. During the interview, we understood that certain activities might be performed on a continuous basis, they might be repeated in different phases of the process, and sometimes, depending on the context, they might never be performed. Therefore, to infuse flexibility into the SeDB model, we decided to use the concept of activity spaces as suggested in ESSENCE (SEMAT, 2013). We feel that our model would be more versatile, that it would better reflect the industrial reality, and that it would allow organizations to easily adapt their models to their specific needs.

Finally, the experience and lessons learned as provided by the academic interviewee gave us substantial feedback for extending our model with process management related issues. Many processes are very difficult to define and manage. Organizations repeatedly make the same mistakes due to the fact that they do not put effort into making the process knowledge persistent and into
making process improvement continuous. For this, reason, we intend to extend our model with database selection process management activities which will help the organizations to learn from the old experience, and which will help organizations to continuously improve their own process models.
Chapter 6. Improved SeDB process model and its Second Round of Evaluation

In this chapter, we present the improved version of the SeDB model and the results of the second round of evaluation with the industrial interviewee. Section 6.1 provides the context of the SeDB model. Section 6.2 describes the model itself, and finally, Section 6.3 presents its industrial evaluation results.

6.1. Endeavor Context and Echo System of the SeDB Process Model

All processes are executed in some particular endeavor contexts and echo systems and a database selection process is no exception. Also, processes may be part of other processes. As shown in Figure 6.1, a database selection process is part of a major process which we call database lifecycle process. By placing the SeDB process model on the lifecycle map, we may clearly see that our contribution is a small part of the overall database lifecycle.

As our academic interviewee pointed out, databases do not live an isolated life. They are part of an echo system, a system consisting of interconnected parts and groups of people who interact with one another. As shown in Figure 6.1, the SeDB’s echo system includes applications that store and retrieve database data, hardware and software platforms on which the database is installed, users using the database and database data, managers and software engineers involved in managing, developing, evolving and maintaining the database, and finally, partners and business(es) that are supported by the database.

All selections of databases do not come from nowhere. They are always part of some bigger endeavor. This is especially true in the context of a database selection process. Today, most organizations already have some form of a database supporting their operation. Very few acquire a totally new database for which they write new applications from scratch. Hence, we suggest that the SeDB process model be part of some other major endeavor, such as reengineering, migration, handover, and the like. However, we do not exclude that it may also be part of a new database acquisition endeavor.

Figure 6.1. Illustrating the endeavor context and echo system of the SeDB process model
If the database selection concerns the acquisition of a totally new database that is not going to be part of any major endeavor, then its process may be somewhat simpler in the sense that companies do not need to pay much heed to many parts of the echo system. Of course, they already have some parts of the echo system implemented such as business, partners and software and hardware platforms that support other non-database or database-based applications. However, they have no applications that need to be rewritten. Here, they have to put their effort into studying what type of future echo system they would like to achieve so that it fits with the current one.

If the database is already part of some major endeavor, then organizations must pay heed to it in great detail. Much business information and rules may be implemented in the old database and its applications. While replacing an old database or migrating to a new one, a minor negligence may have substantial impact on the database acquisition success. Selecting a new database may also imply that the whole echo system must be revised and exchanged. For instance, acquiring a new NoSQL database may imply that all software and hardware platforms must be exchanged, most of the applications must be rewritten, and parts of the business operation must be modified.

6.2. The Improved SeDB Process Model
The Select Database (SeDB) process model provides a set of generic activities that can be chosen on an as needed basis. Its goal is to assist companies in their database selection processes. As shown in Figure 6.2, it consists of three parts. These are (1) activity types, (2) activity spaces, and (3) the SeDB blueprint. The activity types are suggestions for what activities need to be performed in a database selection process. The activity spaces correspond to any major or minor phases that are part of the SeDB process model whereas the SeDB blueprint provides a suggestion for how to arrange the activity types within the activity spaces.

6.2.1 SeDB Activities
The SeDB activities are grouped into twelve groups. As shown in Figures 6.3-6.5, these are (1) CS Determine condition and supportiveness of the currently used database, (2) DBS Determine that a new database solution is needed, (3) MS Manage stakeholders, (4) DI Disseminate information about the new database selection endeavor, (5) D Make a decision, (6) MDB Manage DB selection process, (7) P Plan, (8) SDB Select a database, (9) BV Determine business value, (10) B Assign/reassign budget, (11) R Manage risk, (12) ET Educate and train. The descriptions of the majority of these activities are provided in Appendix C. Hence, they will not be provided herein.
### CS: Determine condition and supportiveness of the currently used database

- **CS-1:** Identify/re-identify (a) problem(s) with the current database.
- **CS-2:** Identify/re-identify underlying root cause(s) to the problem(s).
- **CS-3:** Determine/re-determine consequences of the problem(s).
- **CS-4:** Identify/confirm a need.

### DBS: Determine that a new database solution is needed

- **DBS-1:** Evaluate/re-evaluate current business model.
- **DBS-2:** Evaluate/re-evaluate how the current database supports the current business model.
- **DBS-3:** Determine/re-determine which type of a database is optimal for the business.
- **DBS-4:** Identify/re-identify business opportunity related to the acquisition of a new database (type).
- **DBS-5:** Define/redefine future business model.
- **DBS-6:** Confirm/re-confirm that a new database solution is needed.

### SDB: Select a database

- **SDB-1:** Elicit the requirements guiding the selection of the new database.
- **SDB-2:** Assure usefulness of the requirements.
- **SDB-3:** Define/redefine test cases.
- **SDB-4:** Determine type of a database needed to fulfill the solution.
- **SDB-5:** Select candidate databases to be studied.
- **SDB-6:** Test the candidate databases.
- **SDB-6.1:** Define test cases to be used in the benchmarking process.
- **SDB-6.2:** Set up testing environment.
- **SDB-6.3:** Test databases using the test cases.
- **SDB-7:** Rank the candidate databases using the testing results.
- **SDB-8:** Suggest the optimal database solution that is appropriate for the company.
- **SDB-9:** Suggest alternative database solutions that are appropriate for the company.

### MDB: Manage DB selection process

- **MDB-1:** Define DB selection process model.
- **MDB-2:** Determine what to document & when.
- **MDB-3:** Monitor and control DB selection process model.
- **MDB-4:** Document the DB selection process, if required and necessary.
- **MDB-5:** Document lessons learned, if any.
- **MDB-6:** Improve DB selection process model, if necessary.

### MS: Manage stakeholders

- **MS-1:** Identify/re-identify stakeholders experiencing the need.
- **MS-2:** Identify/re-identify stakeholders experiencing the problem.
- **MS-3:** Identify/re-identify stakeholders that will use the system.
- **MS-4:** Identify/re-identify stakeholders that will be involved in selecting a database.
- **MS-5:** Identify/re-identify stakeholder that will fund the new database and its implementation.
- **MS-6:** Identify/re-identify stakeholder group representatives.
- **MS-7:** Identify/re-identify stakeholders to be informed about the database selection process.

### DI: Disseminate information about the new database selection endeavor

- **DI-1:** Determine communication channels to be used for information dissemination.
- **DI-2:** Determine milestones for information dissemination.
- **DI-3:** Disseminate information, when relevant.

### D: Make a decision

- **D-1:** Study all the data underlying the decision making.
- **D-2:** Study the current and target business models.
- **D-3:** Study the suggested optimal database solution.
- **D-4:** Study alternative database solutions.
- **D-5:** Study the costs of choosing the optimal versus alternative solutions.
- **D-6:** Study the impact of choosing the optimal versus alternative solutions.
- **D-7:** Study the business value.
- **D-8:** Make a decision.
- **D-8.1:** Make a decision whether to proceed with the database selection process.
- **D-8.2:** Make a decision on whether to acquire new database.

---

**Figure 6.4. Groups of the SeDB activity types, part 1**

**Figure 6.5. Groups of the SeDB activity types, part 2**
By studying the SeDB activities, the astute reader should understand that the majority of the activities are of management type. We explain this with the fact that the database selection process is part of the initial phases of some bigger endeavor in which the majority of the frontend activities are rather on a management level whereas the majority of the backend activities are rather on a software engineering level.

### 6.2.2 Activity Spaces

Activity spaces are containers of activities that are to be selected from the activity types and arranged in the order that suits any specific database selection project. The activity spaces may correspond to a whole process or to a phase or a sub-phase. They may be organized in sequences or they may run in parallel. Their content stands for either planned or executed processes. If the activity spaces are empty, then this means that nothing has been planned or executed.
Figure 6.7. The blueprint of the SeDB process mode, 1 part 1
6.2.3 SeDB Blueprint

In the blueprint of the SeDB process model, we suggest thirteen activity spaces. As shown in Figure 6.6, these are (1) Manage SeDB process, (2) Disseminate information, (3) Identify need, (4) Establish current status, (5) Establish target status, (6) Confirm need, (7) Identify DB requirements, (8) Make a feasibility study, (9) Select Database, (10) Make a decision, (11) Plan, (12) Educate and train, (13) Manage risks. The activity spaces are filled in with the activities as outlined in Figures 6.4, 6.5 and 6.6.

The activity spaces correspond to phases in a process. By studying Figure 6.6, the astute reader may see that many of the phases run in parallel with the main body of the phases which are (1) Identify need, (2) Establish current status, (3) Establish target status, (4) Confirm need, (5) Identify DB requirements, (6) Make a feasibility study, (7) Select Database. For this reason, their time window is extended along the whole process cycle.

The differences to the former version of our model lies in the fact that in this version, we allow the activities to be conducted in any order. We also allow them to be included in any process phase. The users of the SeDB process model are free to use them in the order that best suits their businesses, requirements, needs, formality levels, development approaches, contexts and specific working and/or technological environments as long as their choice contributes to the quality of their results.

The astute reader might also notice that we have phases with the same name, however, with different numbers. For instance, the Disseminate Information phase is numbered as 1 and 2. The numbers indicate that the phases have different constellations of their inherent activities. In the Disseminate Information phase 1, one must (1) determine communication channels to be used for information dissemination, (2) determine milestones for information dissemination, (3) identify/re-identify stakeholders to be informed about the database selection process, and (4) disseminate information. In the Disseminate Information phase 2, on the other hand, one has already conducted the first three activities, and therefore, only the disseminate information activity needs to be conducted.

As shown in Figure 6.6, the SeDB blueprint consists of a number of activity spaces. Some of them are used for the entire duration of the process. Others are performed in a sequential order that provides input and output for the adjacent activities. The main body of the activity spaces that have a sequential order are (1) Identify need, (2) Establish current status, (3) Establish target status, (4) Confirm need, (5) Identify DB requirements, (6) Make a feasibility study, (7) Select Database.

In the first phase, one identifies need, problems and consequences and stakeholders involved. The second phase (Establish current status) communicates the need for evaluating the current business model. It incorporates the business model, evaluation of the current database and determination of the business value. The next phase (Establish target status) is similar to the previous one. However, it focuses on the targeted business model including additional activities such as identifying business opportunities, defining the impact of introducing a new database and the impact on the organizational echo system.
After the current and target status states have been evaluated, companies should confirm the need for acquiring a new database (Confirm need). If the need is not confirmed, then companies may continue with the database selection process, and thereby, run the risk of losing effort and resources to later find out that they actually did not have any need.

After the need has been confirmed, one should elicit requirements guiding the selection of the new database and assure its usefulness (Identify DB requirements). This activity also incorporates the determination of the type of databases that are believed to fulfill the need. The following activity (Make a feasibility study) ensures that the project should move forward. Here, one should analyze the...
current database, analyze the usefulness of the requirements and estimate the cost. Then the last step (Select Database) is concerned with defining test cases, testing the databases, and suggesting an optimal database solution.

The activity spaces that are used throughout the lifecycle of the database selection process are (1) Manage SeDB process, (2) Disseminate information, (3) Make decision, (4) Plan, (5) Educate and train, and (6) Manage risks. These activities are continuously used throughout the whole process.

The first phase (Manage SeDB process) refers to the overall definition of the entire process, monitoring and controlling it and improving it, if necessary. It also includes a very important however often questioned activity of documenting the process. To allow freedom of determining process formality, we suggest that each company should determine what should be documented and when. The astute reader might question our choice of mixing process execution with process management. We do it on purpose to indicate the agile trend of self-management, the management in which both managers and developers are involved.

The second phase (Disseminate information) aims at informing all the people involved in or affected by the process and the process outcome. The SeDB model proposes that this activity be performed
three times. The first time is when the need has been identified, the second time is when the need has been confirmed, and the last time is when a database has been selected.

The decision making phase (Make decision) is performed twice in the process model. The first time is when a need has been confirmed and the second time is when a database has been selected. This is to make sure that the data and knowledge gathered thus far shows that a correct decision has been made and that the process should continue as is. Of course, the SeDB process model does not exclude that minor decisions are conducted along the process.

The fourth phase (Plan) is concerned with planning the process after the need has been confirmed. It involves activities such as defining milestones and deliverables, identifying projects constraints and monitoring progress against the plan. Just because planning is an integral part of the modern process models, the SeDB process model suggests that it be conducted along the whole process.

The fifth activity (Educate and train) contains activities such as the identification of stakeholder groups to be educated and trained and determining educational needs for each stakeholder group. The SeDB process model suggests it be performed after the need has been confirmed. Here, it is important to identify stakeholder groups to be educated and trained and to determine their educational needs. At this phase, the group of software professionals that should be educated and trained is the group that is involved in selecting and testing candidate databases. Regarding the education and training of other stakeholder groups, we believe that they are not very much involved in the database selection process. They become more involved during the installation process phase, the phase that follows the database selection phase. However, organizations are free to decide when to educate and train different stakeholders.

The last phase (Manage risks) is performed several times during the process. In its initial phases, it involves activities such as identifying and analyzing risks. In the last phase, it involves planning risk management, monitoring and controlling risks and the continuous update of the risk list.

6.3 Round 2 Evaluation of the SeDB process model

In this section, we present the results of the second round evaluation with our industrial interviewee. In Section 6.2.1, we present the results of the interview whereas in Section 6.2.2, we analyze the interview results. While presenting the results of the interview and their analysis, we follow the order of the evaluation criteria and their respective questions as defined in Chapter 4.

Before being interviewed, the industrial interviewee received our description of the state of art within research that is presented in Section 3.2, the improved SeDB model as presented in Section 6.1 and the questionnaire as described in Chapter 4.

6.3.1 The Industrial Interviewee Credibility

The industrial interviewee deemed that he had enough experience to participate in the interview and to answer all the questions. He is a senior consultant at his own company. He has had this role for over two decades. He has a Ph.D. in computer and systems sciences and has been a lecturer at Stockholm University for ten years. His PhD topic dealt with object database schema design.

After he pursued his PhD, he started working as an IT consultant. During his industrial career, he has participated in several projects mostly at Swedish Defense Material Administration. These projects include project and area management, requirements engineering including data modeling, process modeling, database design, general requirements and acquisition requirements, aircraft baseline coordination, and the like. He has been involved in four database selection processes. These projects
have been in different business areas such as insurance companies, government authority and Swedish defense.

6.3.2 Semantic Correctness
The industrial interviewee was of the opinion that the activities described in the process model were relevant and applicable within a database selection process. He pointed out that many steps were business oriented and he agreed that all of them were important for the project success. Hence, the industrial interviewee did not have any remarks considering the semantic correctness of the SeDB process model. He was of the opinion that the improved version of the SeDB process model had good coverage of the activities.

6.3.3 Syntactic Correctness
The industrial interviewee was of the opinion that there were no redundant or unnecessary steps in the proposed model. According to him, the overall sequence of the steps in the SeDB blueprint was adequate, and thereby, the syntactic correctness was achieved. He was however somewhat confused by one of the activity groups and the difference between them. These were (1) Establish target status and (2) Establish current status. He wondered why there were more activities in one of these, and pointed out that these could be grouped together as one activity group.

6.3.4 Process Flexibility
The industrial interviewee liked the idea of activity groups and activity spaces. He was of the opinion that the improved version of the SeDB process model offered much flexibility. According to him, it would help many practitioners understand that the process is not rigid and that it might be easily adapted to various contexts. He felt that the process model could be used in various database selection projects and that it would definitely be of great help. However, he pointed out that seldom would a database selection process involve all the steps of the SeDB process model. Finally, since the model is very flexible and adaptable for each and every project, the industrial interviewee pointed out that the model should be supported with guidelines. People should understand why and when they should use the SeDB activities.

6.3.5 Usefulness
The industrial interviewee believed that the SeDB process model was a useful tool in guiding companies in choosing a database. He characterized the improved SeDB model as an advanced checklist. However, he also sought after simplicity and lucidity. If a process model is not simple and lucid, it will never be used because nobody will understand it. For this reason, he liked the idea of including the SeDB blueprint. Here, he pointed out the importance of having something concrete like a process model guiding the project in the right direction. But yet again, here he once again pointed out the importance of understanding the process model and the use of it. When being asked about whether he would consider using the SeDB process model, he claimed that he would definitely do it. The constellation of the activities to be used and their order would however depend on the organizational culture, procurement process, and the organizational echo system.

6.3.6 Experience
From the perspective of experience, the industrial interviewee did not see any problems with the SeDB process model as long as it was correctly understood. He believed that lack of standards is a major problem on today’s database market. This both concerns lack of SQL standards and lack of standard process models for selecting databases. Lack of SQL standards creates immense difficulties when companies need to transition from one relational database to another. Lack of database selection
standard process models makes many companies unaware of the tremendous size and scope of the process.

According to our industrial interviewee, the most challenging thing in a database selection process is the cost which is strongly related to the unawareness of the size and scope of the process. The budgeting decisions typically occur on a higher management level in an organization. Management seldom realizes the cost of acquiring new technology and this may jeopardize the entire database selection project. He has experienced this in many projects he had participated in.

6.4 Analysis of the Round 2 Interview

The industrial interviewee has participated in several database selection projects. Also, he has a thorough experience with many process models and methods within software engineering. To add zest to it, he is a PhD in database engineering. Therefore, we deem the industrial interviewee as credible and relevant for the interview.

Regarding the semantic correctness, the interviewee was of the opinion that the steps described in the process model were all relevant and applicable. When it came to the syntactic correctness, the interviewee pointed out that some of the activity groups could have been put together for the sake of simplicity. These groups were (1) Establish target status and (2) Establish current status. Although they have many similar activities, we chose to have them as separate groups because these activities have different goals and different sets of activities. By merging them into one group, we feel that we would lose the expressiveness of the SeDB model. However, we may suggest that one may create a more general group for these two activities. We believe that by keeping them as they are right now, we will help the organizations to distinguish between those two, and thereby, establish better monitoring and control over the process.

The industrial interviewee thought that the overall structure and flexibility of the model was good. He wished that the SeDB model be complemented with guidelines. We agree with the industrial interviewee that process guidelines are needed to make the process more understandable and less prone to error with regards to each process instance. They would help people know when and how certain activities should be used. Guidelines could make the process simpler and more lucid which was the final remark of the interviewee.

We feel that adding guidelines at this very late stage of the master thesis would imply embarkment on uncharted waters. The scope of our thesis is huge and providing guidelines at this stage may cost us as much effort as to write a new master thesis. For this reason, we will consider the industrial interviewee’s suggestion in our suggestions for future work.
Chapter 7. Benchmarking

In this chapter, we describe the benchmarking process, the process that was part of the evaluation of the SeDB process model via action research. We first describe the context of the robots to be benchmarked and benchmarking requirements as stated by Comau via CNet in Section 7.1. We then describe the benchmarking process as we conducted it in Section 7.2.

7.1 The Benchmarking Context

As a manufacturer, Comau uses robots for producing automotive vehicles. These robots consume energy and Comau wishes to find out how much. To be able to do it, Comau collects real-time data on the energy consumption from the robots. These data are collected five times per second and they communicate the nature of electricity and an alternating current (AC), a periodical reverse of the conductor or semiconductor movement.

Robots demand electricity in one way or another. In many cases, however, their electrical need may be less than the supplied electricity. Therefore, in an AC circuit, there are several values that determine the electrical need of a certain device. These are True Power (P), Reactive Power (Q), Apparent Power (S), and Power Factor (PF).

Comau explicitly requested to collect data on P, Q, S and PF values. For each of these values, they in turn requested that different sub-values be collected every 200 milliseconds. As illustrated in Figure 7.1, these values are P1-P3, Q1-Q3, S1-S3 and PF1-PF3. Unfortunately, we did not receive any explanation of what was meant by those individual values. We did not have any contact with Comau and the interviewees from CNet did not know it. Neither could we find any explanation on the internet. However, we suspect that they stand for electricity as consumed by different parts of a robot.

Using the above-described context of the robots, we summarize the overall input to be used in our benchmarking process. It is the following:

One robot produces five records displaying electricity usage during one second interval. This implies that when 200 milliseconds pass, a new series of data is produced. Each record includes values P1, P2, P3, Q1, Q2, Q3, S1, S2, S3, PF1, PF2 and PF3. Each robot record is identified by a robot ID and a timestamp.

7.2 Benchmarking Process

The benchmarking was performed at CNet. The goal was to implement the latter part of the SeDB model by conducting action research. It covered the following SeDB phases and activities as selected from the SDB: Select a Database activity group:

- SDB-1: Elicit the requirements guiding the selection of the new database.
- SDB-4: Determine type of a database needed to fulfill the solution.

Figure 7.1. Illustration of Comau Robot and the data it produces
- SDB-5: Select candidate databases to be studied.
- SDB-6: Test the candidate databases.
  o SDB-6.1: Define test cases to be used in the benchmarking process.
  o SDB-6.2: Set up testing environment.
  o SDB-6.3: Test databases using the test cases.
- SDB-7: Rank the candidate databases using the testing results.
- SDB-8: Suggest the optimal database solution that is appropriate for the company.

Those steps are described in Sections 7.2.1 – 7.2.4, respectively.

7.2.1 Eliciting Benchmarking Criteria

As a first step, we gathered requirements for selecting a database for IoT projects. We did this while conducting initial interviews. The questionnaire used is presented in Appendix B. For the benchmarking purposes, only Questions 8–13 were relevant. With those questions, we received good understanding of Comau’s needs and requirements. As shown in the upper part of Table 7.1, the requirement types as stated by Comau were:

- (SR1) Ability to store x number of metric values per x number of robots per time unit. This requirement type was needed because a robot produced five metric values per second. Comau wished to have a solution that would view all the data points from a particular robot. If all the data points were not collected in time, then the data gathered would be misleading.
- (SR2) Ability to perform aggregation with the x number of metric values for one robot per time unit. This requirement type concerned gaining knowledge from the data collection. Comau wished to use the data for further analysis of the robots’ energy consumption. Aggregating these data could provide useful information for optimizing the energy consumption in the whole manufacturing process.
- (SR3) Ability to continuously store and present real-time data per time unit for one robot without any delay. This requirement type was needed for getting an immediate status report from the robots. Comau wished to view data in real-time in order to find robots that might be in need of maintenance due to irregular electrical use. The data viewed in real-time could identify robots that were not performing as expected.
- (SR4) Ability to continuously store and present real-time aggregate data for one robot per time unit without any delay. This requirement type was needed for getting an immediate status report from the robots. Comau wished to view aggregated data in real-time for further analysis of one robot’s energy consumption.

The time units are one second and ten seconds. The number of robots is one robot, ten robots and one hundred robots. When studying those four requirement types, we may easily see that all of them may be classified as three groups of requirement types:

- **Functional requirements** communicating the function of storing, aggregating, and presenting real-time data as gathered from the robots.
- **Non-functional requirements dealing with performance** communicating the fact that the database must be able to store the real time data within a predetermined time unit.
- **Non-functional requirements dealing with scalability** communicating the fact that the database may increase the number of robots from which it will collect data.
Table 7.1. Benchmarking requirements used for database selection. SR stands for *Stated Requirements* whereas AR stands for *Additional Requirements*

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SR1</td>
<td>The database has the ability to store x number of metric values per x number of robots per time unit</td>
</tr>
<tr>
<td>SR2</td>
<td>The database has the ability to perform aggregation with x number of metric values for one robot per time unit</td>
</tr>
<tr>
<td>SR3</td>
<td>The database has the ability to continuously store and present real-time data per time unit for one robot without any delay</td>
</tr>
<tr>
<td>SR4</td>
<td>The database has the ability to continuously store and present real-time aggregate data for one robot per time unit without any delay</td>
</tr>
<tr>
<td>AR1</td>
<td>The database is accepted and used by a large number of users</td>
</tr>
<tr>
<td>AR2</td>
<td>The community developing and evolving the database is stable</td>
</tr>
<tr>
<td>AR3</td>
<td>The database supports the Internet of Things projects</td>
</tr>
</tbody>
</table>

7.2.2 Determining Type of Database

Comau wished us to evaluate NoSQL databases. However, to assure that the right database type got chosen, we started studying all different types of databases. Our choice agreed with Comau’s choice. We deemed NoSQL databases to be the most appropriate for this case study. We motivate this with the aggregated nature of the NoSQL data models. These are highly flexible and offer unparalleled scalability for large data (Manoj, 2014; Sadalage, Fowler 2013).

7.2.3 Eliciting Additional Database Selection Criteria

After having determined the type of databases to be studied, the next step after would be to choose databases to be benchmarked. At this moment, however, we realized that it was too early to do it. NoSQL databases are relatively new technologies, and there is not yet any formal definition of them. To add zest to it, there are over 150 different NoSQL databases and the majority of them are open source. Hence, choosing between them can be very hard. To our despair, the company gave us complete freedom in selecting them on our own. For this reason, we defined additional selection criteria that would guide us in the selection process. As shown in the lower part of Table 7.1, these were:

- *(AR1)* **Popularity - the database is popular among a large number of users.** We define popularity as a state of being accepted by a large number of users. Popularity is essential. It assures that the organization is not the only one that invests in the database. Open source communities do not provide support contracts in the same way as proprietary vendors do. Investing in open source databases may imply that a company will not be provided any contracts with service level agreements. Hence, to choose any database may be risky. Therefore, one should evaluate the popularity of the database. If the database is popular, then it is used by many users, and therefore, there is a chance that the open community responsible for it will feel encouraged to further evolve it. It is difficult to find the exact number of users. Hence, to determine the popularity, we tried to find out the spread of using the database within all types of software user community such as industry and academia.

- *(AR2)* **Community - the community developing and evolving the database is stable.** We define community as a group of people who share the common interest in developing and evolving
NoSQL databases. This requirement helped us determine the status of the database supplier community. In open source investments, communities are essential if problems are encountered. The fact that a database is popular does not assure that it will be available for years to come. The community may dissolve, and hence, the database may no longer get evolved. At its worst, companies acquiring a database from such a community run the risk of either having to search for a new database or of having to evolve and maintain it by themselves. To find out the stability of the community, we searched for websites of various database communities. Websites with rich content with up-to-date links and fora were deemed as solid communities.

- **(AR3) IoT - the database supports the Internet of Things projects.** This requirement was used to ensure that the database was suitable for the Internet of Things projects, that is, that it was capable of storing data from electronic devices in order to exchange it with the manufacturers or developers. To fulfill this requirement, we searched for databases that had been used in similar projects with a similar goal.

### 7.2.4 Selecting Candidate Databases to be Benchmarked

At the Select candidate databases to be tested activity, we selected the candidate databases using the three selection criteria (*Popularity, Community* and *IoT*). When identifying the popularity of NoSQL databases, we studied books, articles and various webpages in which most of the authors explained certain data models by referring to a certain database. For example, if an article explained the document data model, the database MongoDB would be used as an example. This helped us create an initial list of the databases that would be further evaluated.

When identifying the community responsible for the database, we studied their webpages and possible community webpages associated with the database. MongoDB proved to have the strongest and largest community whereas the other database communities were less predominant. Finally, when it came to IoT, we studied whether the databases were used in the IoT projects.

Using the *Popularity* and *Community* selection criteria, we chose four NoSQL databases for further evaluation. These were MongoDB, HBase (*OpenTSDB*), Cassandra, Azure Table Storage (*Azure Table Storage, 2015; Cassandra, 2015; MongoDB, 2015; OpenTSDB, 2015). However, after having used the third criterion, the IoT criterion, we found out that *Azure Table Store* was the worst option. We could not find any evidence that Azure Table Store was used within the IoT projects. Despite this, we included it in our candidate list. The decision was based on the fact that CNet, our commissioning company, had a license for using it.

Initially, we planned that the four databases would be sequentially selected for being tested within the time assigned to us, that is, ten weeks. At the time of selecting them, we did not know whether time would allow us to test them all. Therefore, we decided to test one database at a time. To assure that we managed to test the most pertinent databases, we prioritized them according to the evaluation results (AR1-AR3). We arrived at the following priority list:

- **Priority 1 – MongoDB:** MongoDB proved to be the most popular database. Several books use MongoDB as an example of document data stores. It is used by many companies such as LinkedIn, Adobe, ebay and Bosch. Bosch, for instance, uses MongoDB as an underlying architecture for its IoT projects. Furthermore, the MongoDB community proves to have many dedicated contributors. It has as many as 35 000 members and at the moment of writing this thesis as many as 46 000 *Stack Overflow* questions were posted.

- **Priority 2 – HBase (*OpenTSDB*):** HBase proved to be a popular database as well. It is also used in books for describing column-family data stores. It is widely used by companies such as
Facebook, Yahoo and eBay. HBase has been developed for time series data by using a lightweight application called OpenTSDB. This application proved to be gaining ground in IoT projects. Its website provides extensive documentation. In contrast to MongoDB, the size of the community surrounding OpenTSDB is however unknown. For this reason, we give it priority 2.

- **Priority 3 – Cassandra**: Cassandra is also a column-family data store that is used as an example in books and articles. The database is used in companies such as Netflix, Instagram, Accenture and Spotify. Successful stories of using Cassandra within IoT projects have been published and the community responsible for Cassandra has proved to be very active (Planet Cassandra, 2015). Cassandra is in many ways similar to HBase. Because of HBase’s extensive documentation, we chose to give Cassandra priority 3.

- **Priority 4 – Azure Table Storage**: Azure Table Storage is a key-value data store that is developed by Microsoft. It is a NoSQL database that has a community built around it and Microsoft is dedicated to keeping the Azure Table Storage around for a long time to come (Azure Table Store, 2015). However, it is not used by many companies today. Azure Table Storage is not open-source and it can only be used in the Azure cloud platform. For this reason, we assign it priority 4.

### 7.2.5 Testing the Candidate Databases

To test one database implies defining test cases for the database under consideration, setting up a testing environment and then testing it. Within the time assigned to us, we managed to test only two databases. These were MongoDB and OpenTSDB. This section reports on our testing results. Section 7.2.5.1 briefly describes the required output formats and explains why different test cases had to be defined for each of them. Section 7.2.5.2 briefly presents the test cases. Section 7.2.5.3 presents the testing environment together with the testing process and its results.

#### 7.2.5.1. Testing Output Formats of MongoDB and OpenTSDB

The output from the tested robots resulted in two types of records. The first type included records listing all the AC values \((P_1, P_2, P_3, Q_1, Q_2, Q_3, S_1, S_2, S_3, PF_1, PF_2\text{ and } PF_3)\) (see Section 7.1) whereas the second output included the result of a specific test case for a particular instance of the requirement type \((SR_1, SR_2, SR_3,\text{ and } SR_4)\) (see Section 7.2.1). Summing up, for all test cases, each database produced two results: (1) **Output 1** consisting of records including the AC values as produced by the robot \((P_1, P_2, P_3, Q_1, Q_2, Q_3, S_1, S_2, S_3, PF_1, PF_2\text{ and } PF_3)\) and (2) **Output 2** consisting of the values that were required by a particular requirement. Here, **Output 1** constitutes input to **Output 2**.

Regarding **Output 1**, it differs in the two databases tested. Having different data models, MongoDB and OpenTSDB produced different records. The differences may be clearly discerned by studying the left hand sides of Figure 7.2. As illustrated there, MongoDB data stores all its values in one and the same record whereas OpenTSDB only records one AC value per record thus resulting in twelve separate database records.

Summing up, **Output 1** is input to **Output 2**. **Output 1** stores all the values as gathered from the robot. Its format is common for all the tests that were conducted in this study. It only differs with respect to the number of records included in it. **Output 1** is not enough for evaluating whether a particular requirement instance got fulfilled. More compilation needed to be done. To adapt it to the tested requirement at hand, our test cases transformed **Output 1** into **Output 2**. **Output 2** varied for almost all the tests depending on what was being tested. It ranged from simplest cases where only one of the AC values was displayed from a certain robot to more complex cases where aggregated data were continuously displayed in real-time. All outputs representative for all test cases are illustrated in Figures 7.2 and E.1.
Figure 7.2. Illustrating Output 1 and Output 2 from MongoDB and OpenTSDB. MongoDB output is illustrated on the top, OpenTSDB on the bottom.

Output 1 is an intermediary testing result whereas Output 2 is the final testing result. When presenting the output of the test cases, only Output 2 will be used. This output is common for the two database tested.

By studying the amount of requirement instances, the number of values to be tested and the number of various outputs to be produced, the astute reader may suspect that testing all those values was voluminous and complex.

Figures E.2 and E.3 in Appendix E lists the test cases to be conducted on only one database. Altogether, they amounted to 87 test cases in their number. Considering the fact that two databases were tested, we conducted 174 tests altogether.
7.2.5.2 Defining Test Cases
For each of the requirements (SR1-SR4) as stated by the commissioning companies and defined in Section 7.2.1, we created test cases. Altogether, we created 87 test cases that were executed for each database. This amounts to 174 test case executions. Due to space restrictions, we cannot present and describe all the test cases herein. We do it in Appendix E instead.

7.2.5.3. Test and Benchmarking
Even though the two databases had very different data models, they both proved to be suitable candidates for this action research. When being tested, both of them produced the required testing results. As shown in Table E.1 (in Appendix E), MongoDB presented the correct output for all the test cases. OpenTSDB, on the other hand, presented the correct output for all the test cases except for a subset of test cases (52-63) implying that OpenTSDB could not view data with millisecond precision. Although OpenTSDB has integrated the possibility to store timestamps with millisecond precision, it did not allow these values to be presented with that precision. For only this reason, MongoDB was suggested to Comau as an optimal solution.

7.3. Evaluation of the Adherence of the Benchmarking Part of the SeDB Process Model
While conducting benchmarking, we strictly followed the three evaluation criteria that were relevant for this evaluation part and the parts of the SeDB process model. The evaluation criteria were semantic and syntactic correctness and usefulness.

It goes without saying that all the evaluation criteria got fulfilled. By studying Section 7.2, the astute reader should notice that each of the activities was applicable and relevant within the benchmarking process, and thereby, contributed to the quality of its results. The same applies to syntactic correctness. No activity whatsoever was redundant and we did not have any need to create any new activity that was not part of the SeDB process model. All the activities were useful and provided value to the outcome of our results.

The benchmarking process helped us confirm that some activities need be repeated within the process. After having elicited the requirements from the commissioning companies and after having chosen the type of database, we realized that we had to repeat the activity of eliciting requirements. We found out

![Figure 7.3. Old and new version of activity SDB-1](image-url)
that the initial set of requirements would not help us in the benchmarking process. When looking at the SeDB process model, we found out that the model did not cover this context.

As shown on the left hand side of Figure 7.4, the improved version did not cover the re-occurrence of the *Elicit the requirements guiding the selection of the new database* activity (Activity SDB-1). Although not initially planned to evaluate process flexibility, we may conclude that at this phase, we encountered flexibility problem. To remedy this, we reformulated the activity into *Elicit/re-elicit the requirements guiding the selection of the new database*. 
Chapter 8. Analysis and Discussion

In this chapter, we present, analyze and discuss our results. Section 8.1 briefly describes the results of the third round interview. Section 8.2 analyzes the results of all the three rounds of interviews. Finally, Section 8.3 discusses our results from the validity and ethical perspectives.

8.1. Third Round Evaluation of the SeDB process model

The results of the first two rounds of interviews were very different with regards to the interviewees’ feedback. The academic feedback was thorough and her feedback helped change the overall design of the entire process model. The industrial feedback, on the other hand, did not result in any major changes. The reason to this was the fact that the interviewees received different versions of the SeDB process model. They did not evaluate the same model. The academic interviewee evaluated the initial version of the SeDB process model whereas the industrial interviewee evaluated its improved version.

To put both interviewees on equal terms and provide them equal opportunity to evaluate the two versions of the model and its progression, we created a third round evaluation phase which we called Evaluate SeDB Round 3 (please see Figure 2.2 in Chapter 2). The goal was to make results consistent with the feedback received within the first two rounds of the evaluation interviews. Here, we conducted additional interviews where the two interviewees could confirm that the improved version was a better solution than the initial one.

Both interviewees were asked additional questions whose goal was to confirm that the progress had been made in the right direction. The academic interviewee was asked whether the new SeDB process model got improved from the perspectives of semantic correctness, syntactical correctness, usefulness and process flexibility. The industrial interviewee, on the other hand, was asked a similar question, however, from the perspective of the initial SeDB process model. He was now presented its initial version. Additionally, both interviewees were asked whether the latest version of the SeDB model was a satisfactory improvement.

The academic interviewee was delighted to see such substantial changes to the initial SeDB process model. Overall, the semantic correctness was satisfactory and she pointed out that the coverage of the activities had substantially improved. She was glad to see them all. All of the new activities were relevant and applicable. Regarding the industrial interviewee, he was of a similar opinion. According to him, the initial model was semantically correct, however, its improved version was much better.

Both interviewees were of the opinion that both models were syntactically correct. However, the industrial interviewee pointed out that some of the explanations of the new steps were weaker than others. Despite this, we decided not to further elaborate on them. It takes time to elicit theory about them and this would strongly expand the scope of our research. Instead, we will suggest it as a future work. So far, we only provide a brief overview of what they entail.

The academic interviewee claimed that the semantic correctness was fully satisfied for the type of academic work and time window assigned to the master thesis. She could not find any redundant or unnecessary steps in the improved SeDB process model. Regarding the missing steps, she claimed that it was almost impossible to provide full coverage of the process activities due to the fact that processes are very complex and comprehensive. In the industry, there are special process experts who spent years on defining processes and, out of what she had seen, those processes are many times not as good as the SeDB process model. Finally, she was of the opinion that the sequence as proposed in the SeDB blueprint was satisfactory. Here, she accentuated the importance of having clear guidelines and rules for helping organizations with defining their own processes.
Regarding the industrial interviewee, he was of the opinion that even the initial SeDB process model was satisfactory. However, having seen its improved version, he claimed that the initial model felt somewhat “clumsy” in its rigidity. Although it did not have any redundant or unnecessary steps, its proposed sequence was too traditional and too much prescribing. By comparing the two versions, he claimed that one could see that many of the important activities in the improved model were not covered in the initial model.

Both the interviewees were of the opinion that the process flexibility got substantially improved. Both of them also advocated guidelines for how and when to use the activities. They both stated that the SeDB activity spaces and the blueprint complemented each other in an excellent way. The academic interviewee however wished more blueprints, or more scenarios as she expressed it. Being adapted to various contexts, the scenarios might be useful for smaller companies.

Despite the huge differences between the two model versions, both interviewees were of the opinion that the two versions were useful, however, the improved version was much more useful and much more versatile. After having analyzed the two, they claimed that the final SeDB process model was a satisfactory improvement.

As the industrial interviewee pointed out, the usefulness lied in the list of activities that need be done. Even if the initial version was scarcer in their amount, still however, it provided good roadmap of activities that needed to be conducted. According to him, the usefulness lied in providing feedback on what needed to be done. The academic interviewee was of a similar opinion. The improved version was much more useful not only within the industry but also within the education. It makes its user aware of the vast scope of the process domain.

8.2. Continued analysis

In this section, we analyze the results of the three rounds of evaluation. When analyzing the interview results, we follow the order of the evaluation criteria: (1) interviewee credibility, (2) semantic correctness, (3) syntactic correctness, (4) process flexibility, (5) usefulness, and (6) experience. When evaluating the benchmarking results, we follow the three criteria (1) semantic correctness, (2) syntactic correctness, and (3) usefulness. However, due to the experience gained during the benchmarking process, we accidently evaluated the model’s flexibility.

We had the luxury of interviewing two highly skilled professionals both within the academia and the industry. Both of them had a doctoral degree in software versus database engineering and both of them had at least a two decade experience in industrial software processes. Hence, we deem them as highly motivated professionals to evaluate the SeDB process model.

Before the Evaluate SeDB Round 3 phase, our interviewees had different prerequisites. The first interviewee was provided the initial version of the SeDB model as presented in Section 5.1. Because of the extensive feedback received, we may claim that the first interview might be classified not only as evaluative but also as explorative. The academic interviewee came up with many improvements that were mostly based on an OMG standard called ESSENCE (SEMAT, 2013). This implies that the second interviewee was served a ready SeDB process model “on a silver platter”.

Both interviewees were of the opinion that the process steps in the improved SeDB process model were relevant to be part of the database selection process. According to them, all of the steps are applicable and necessary. The academic interviewee was glad for the fact that we implemented identification and confirmation of need in our model. This will definitely help many companies
eliminate the risk of acquiring database based on personal preferences and wishes. Also, the benchmarking process showed that the activities used were all relevant and applicable.

The industrial interviewee, on the other hand, agreed that the improved version of the SeDB process model was much better. It had a much better coverage of the highly relevant and applicable activities. Also, the benchmarking process proved that all the activities that we chose were relevant.

Both our interviewees were of the opinion that there were no redundant or unnecessary steps in the improved proposed model. The overall sequence of the steps was adequate, and thereby, the syntactic correctness was achieved. The industrial interviewee was however confused with respect to the activity groups and the difference between them. These were (1) Establish target status and (2) Establish current status. He wondered why there were more activities in one of these, and pointed out that these could be grouped together as one activity group.

The new SeDB activities were added thanks to feedback from the academic interviewee. It was she who pointed out lack of planning, identification of problems and their consequences, identification of the stakeholders involved, determination of a type of a database, and accountability. These steps have now been added to the model. Also, the benchmarking process showed that the order chosen in the process was correct. No activity was redundant and no activity was missing.

The most extensive contribution was added in terms of process flexibility. The new design of the SeDB process model changed the entire appearance and use of the process model. This makeover was based on the process model standard called ESSENCE (SEMAT, 2013). It incorporated activity groups, activity spaces and a blueprint for the process model. The initial version of the SeDB model was very sequential and did not leave room for flexibility. The second interviewee agreed that the makeover of the initial process model made the SeDB process model more flexible and that the model would definitely be more applicable in various organizational contexts. Finally, the academic interviewee found it very important to put the process model in the context of a company setting in order to find out how it related to other parts of an organization.

While executing the benchmarking part of the SeDB process model, we got the opportunity to evaluate the model from the flexibility perspective. Initially, we did not expect flexibility to be an important criterion when evaluating such a small part of the model. We were wrong on this part. While following the process, we got a proof that activities may be conducted in any order when need arose. In our case, this need concerned the repetition of the activity whose purpose was to elicit requirements for the databases to be selected. It showed that the first execution of the requirements elicitation did not provide enough feedback. Hence, we needed to repeat the activity. This repetition was not built into the model. We remedied this by reformulating the activity ***SDB-1: Elicit the requirements guiding the selection of the new database*** into ***SDB-1: Elicit/re-elicit the requirements guiding the selection of the new database***.

Regarding the usefulness of the SeDB model, both interviewees believed that it was useful. It constitutes a good basis for achieving a common understanding of what activities must be included in a database selection process. Both interviewees would consider using them in education as well as in an industrial setting. However, the industrial interviewee emphasized that process models need be correctly understood in order to be useful. A process model that is not easily understood will never be used. Both the academic and industrial interviewees agreed that both the blueprint and activity groups substantially increased the usefulness of the model. However, to more increase the models usefulness, both of them sought after more examples, or blueprints that were adapted to various organizational
contexts. Regarding the benchmarking process, all the activities chosen for the process proved to be useful. All of them contributed to the choice of a database.

Another suggestion for increasing the model’s usefulness came from the academic interviewee. Here, she suggested that the process model be more adapted to the agile trend of continuous and iterative process planning, process management, risk management, education and training, decision making and information dissemination. We considered this in our new model version by suggesting activity spaces that were running along the whole SeDB process model. Finally, we increased the usefulness of the model by placing it in the endeavor process context and echo system context. By doing so, we believe that we have made it clear that a database selection process model does not live an isolated life. It is always part some other endeavor such as, for instance, migration of the whole database-based system to a new hardware and software platform, and therefore, it will always impact the overall echo system.

When being asked about whether the interviewees would consider using the SeDB process model, they claimed that they would definitely do it. The constellation of the activities to be used and their order would however depend on the organizational culture, procurement process, and the organizational echo system.

The experience as provided by the interviewees provided substantial feedback for the SeDB process model extension. As a reaction to the academic interviewee’s feedback on process knowledge, we created an activity space corresponding to the phase of education and training. In response to her feedback on both lack of process knowledge and undocumented and forgotten process lessons learnt, we created the activities that dealt with documentation.

Regarding the industrial interviewee’s feedback on lack of standards, we believe that by creating the SeDB process model, we will be able to increase companies’ awareness of the tremendous size and scope of the process, and thereby, improve their understanding of the process cost. The number of the activities included in our model talks for itself and does not require any further explanation.

8.3. Discussion

In this section, we discuss our results from the validity, ethical perspective and hypothesis perspective. Section 8.2.1 discusses our research from the ethical perspective. Section 8.2.2 discusses our research from the validity threat perspective and Section 8.2.3 motivates why we have arrived at our tentative hypothesis.

8.3.1. Ethics

While conducting our research, we obeyed four ethical requirements. These were (1) information requirement, (2) consent requirement, (3) confidentiality requirement and (4) usufruct (Denzin, Lincoln 2011) As a researcher, we took full responsibility for the results obtained by considering both the organizations and the individuals involved in our study.

The information requirement discloses the need to inform respondents about the research, its purpose and the rights associated with their participation. There was no need to follow this rule with the organizations supporting our research, that is CNET and Comau. Regarding the interviewees, we fulfilled this requirement twice (1) while contacting the interviewees and receiving their agreement for being interviewed, and (2) while inaugurating our interviews, the interviewees confirmed that they wished to be interviewed.
The consent requirement refers to the consent of all the people involved to participate in the research process by being interviewed. The interviewees were informed about their participation and they had the right to discontinue the interview if they wished so. No one however did it.

The confidentiality requirement refers to the right of every party involved to stay anonymous. Both our interviewees wished to be anonymous. Regarding the organizations involved, CNet and Comau, they did not request any anonymity.

Finally, usufruct refers to the research outcome only being used for the intended purpose. In our case, the intended purpose is the creation of the SeDB process model. The material collected during our research was not utilized for any commercial purpose outside this research.

8.3.2. Validity of our results
Validity is concerned with the evaluation of a chosen research method in terms of its appropriateness, strength and soundness. Because of the qualitative approach, this research was exposed to validity threats such as credibility, transferability, dependability, and conformability. They are correspondences to internal validity, external validity, reliability and objectivity in quantitative research.

Credibility within qualitative research corresponds to internal validity within quantitative research. It determines the believability and trustworthiness of the respondents and their answers. In order to deal with credibility, we used the evaluation criteria Interviewee credibility to ensure the trustworthiness of the answers provided. This criterion helped us ensure that the respondents had sufficient experience to be able to provide credible feedback to the SeDB process model.

Transferability within qualitative research corresponds to external validity and it refers to what degree the study’s findings are generalizable beyond the scope of the research. We verified our research findings by validating the data from more than one source. The interviewees were selected from both the industry and academia and they possessed different angles of experience. This assures that the study’s findings may be regarded as generalizable enough, or rather, transferrable from one context to another in the qualitative research parlance.

Dependability within qualitative research corresponds to reliability within qualitative research and refers to the repeatability of the research process. Qualitative research results cannot be replicated in the same way as quantitative research results because of the strong process variation in different contexts. If we were to measure the same thing again, we would measure two different things. Hence, the dependability of the research needs to be proven by repeating the research in different industrial contexts.

Conformability within qualitative research corresponds to objectivity within quantitative research and refers to the degree to which the outcomes could be confirmed by other people. This was ensured by thoroughly transcribing the interview answers and by sending them for confirmation to the interviewees. In this way, we mitigated the risk of unintentional subjectivity, that is, by providing interview results that we might misunderstand.

8.3.3. Arriving at the tentative hypothesis
Our research was of design research paradigm being strongly governed by inductive reasoning along the whole process. Here, we first observed and explored the domain of database selection process by studying literature and by interviewing experts. This, in turn, helped us detect patterns of the SeDB process model and formulate our tentative hypothesis which we then evaluated via interviews and
action research. All this aided us in developing a theory underlying a database selection process. Therefore, we dare claim that “the suggested SeDB process model is a valid solution for addressing lack of database selection process models today”. We arrived at this hypothesis by evaluating the SeDB process model both within the academia, industry and via action research.
Chapter 9. Conclusions and Future Work

Today, all business is run by processes and there are many of them. Some examples are software development processes, maintenance, support and business processes, software acquisition process and the like. Processes are an important ingredient in all companies, and if properly defined and followed, they provide substantial support for the organizations. They also help them monitor and control their operation and discover and remedy various risks. In this chapter, we make final conclusions and suggestions for future work.

9.1 Conclusions

Various standards have been created with the purpose of aiding organizations with defining their processes. Right now, there are many standard models for software development and maintenance. However, there are no standard models for a database selection process. This is very surprising bearing in mind the fact that there are several thousands of databases to choose among on today’s market. When searching for the standard models in well known research reference sources such as IEEE Xplore, ACM, Springer and Wiley (ACM, 2015; IEEE, 2015, John Wiley, 2015; Springer, 2015), we have not found any process model guiding organizations in choosing a database that is appropriate for their business context. This made us realize that there were no published database selection process models. This also made us formulate a research question inquiring about what activities should be included in a database selection process and how they should be organized.

To address our research question, we delved into the unexplored domain of database selection process, elicited theory about it which we then materialized in a database selection process model. The model is called the Select Database (SeDB) process model. The SeDB process model only deals with the selection of the database and does not cover the implementation and installation of the database. The model aims at guiding organizations in selecting an appropriate database for their specific needs and at providing bases for future research. Due to lack of theory on database selection processes, we could not base our research on somebody else’s research results or other models. We had to elicit all the theory from scratch by studying similar process models and by interviewing two experts within the subject.

The research was qualitative and explorative which strictly followed the “design science” paradigm that was governed by inductive reasoning (Johannesson, Perjons 2012). Data collection was primarily conducted via literature study, interviewing, and action research. Data analysis was conducted via the criteria that have been suggested for evaluating process models and via hermeneutics. The selection of the interviewees was conducted using the convenience sampling method with pre-defined selection criteria.

The interview results have shown that the SeDB process model is realistic, that its process steps had semantically accurate meaning and that the model was syntactically correct. No activity in the model felt redundant. According to our interviewees, the model covered an exhaustive list of activities. The interviewees liked the concept of using activity spaces to express process flexibility and the idea of providing an exemplary process model. Both of them agreed that many organizations would benefit from our process model. However, they expressed great concern regarding the understanding of the model by unexperienced practitioners. As a remedy, they suggested guidelines providing directions when, where and by who the activities should be conducted.

Both the interviewees agreed that the SeDB process model was useful and that they would definitely use it in their work contexts. Finally, the feedback from the interviewees on their experiences gained
helped us extend the SeDB process model with the *Educate and Train* and *Manage SeDB process* activity groups.

The action research results have also helped us confirm that the subset of activities being studied was realistic, useful and that it reflected the reality of our benchmarking process. All the benchmarking activities were correct and no activity felt redundant whatsoever.

The feedback from the interviewees and action research and the inductive character of our work (Spohn, 2013) had helped us arrive at a tentative hypothesis stating that the SeDB process model was a valid solution for addressing lack of database selection process models. We dare claim this thanks to the fulfilled evaluation criteria that were (1) **interviewee credibility**, (2) **semantic correctness**, (3) **syntactic correctness**, (4) **process flexibility**, (5) **usefulness**, (6) **experience**.

### 9.2. Suggestion for Future Work

Even if the SeDB process model is a valid solution at the moment of writing this thesis, we agree with our interviewees that it should be further elaborated and evaluated. We believe that process models of this type may mainly be elaborated and evaluated with the aid of qualitative studies. Process models cannot be quantitatively evaluated by using clear cut hypothesis and by generating numerical data. The outcome would not make any sense. We motivate this with the fact that defining and improving process models always takes place in the contexts in which humans and human understanding and beliefs are involved. Process models will always be based on what is known and believed today. Every time, one works with the process, one gets more insight into different situations and problems which lead to the improvement of the process model.

An important issue that was raised during the interviews was accountability. The SeDB activities as they are defined now do not provide much information on who does what. Companies and their employees should feel obliged to account for database selection activities. To do that, one must define roles and responsibilities, assign them to individuals across the organization, and assure the process transparency. For this reason, we suggest that the SeDB process model be extended with the definition of roles and their responsibilities.

![Figure 9.1. Operational levels and their generic responsibilities (Business Analysis, 2015; Management Levels, ud)](image-url)
To define roles is a very difficult task considering the fact that all organizations differ with respect to role names, company size and organizational levels. At its minimum, organizations may have just a few employees, whereas, at the maximum, the organizations may have hundreds of thousands of employees. Also, there is no clear standard what responsibilities should be possessed by particular roles, and even if some roles are implicitly understood, their borders may be very fluid and vary from company to company. For instance, an architect in one company may be responsible for both the system architecture and the processes around it, whereas in another company, the same role may be responsible for the architecture only. Still, however, we believe that the SeDB process model would substantially benefit if it were extended with roles.

It is not enough to only define roles. One must place them on different organizational levels. Figure 9.1 shows a roadmap of three different organizational levels. According to the interviewees, the model would substantially benefit if we placed different roles and responsibilities on this roadmap. Hence, we suggest that the roles and responsibilities be placed on the three levels as outlined in Figure 9.1.

We suggest that the SeDB process model be extended with guidelines, rules and various contextual scenarios. Guidelines and rules should provide clear directions of what needs to be done, when and by who. They should also point out risks that may be encountered if they are not followed. The contextual scenarios are important because they increase the understanding of the process and help practitioners to mature with respect to which activity should be chosen and why.

Finally, we suggest that our activity groups be transformed into practices*. Right now, in modern software engineering, software processes are constructed from a set of practices. In this thesis, due to time and space restrictions, we did not suggest any practices. However, we believe that our activity groups constitute a good platform for defining practices in the future.

### 9.3 Epilogue

Even if software engineering is more than five decades old, it is still an immature discipline (SEMAT, 2013). It possess many gaps in form of undocumented knowledge and experience about various software engineering processes and it lacks process models for many software engineering endeavors. One of them is database selection process model. To add zest to it, the software community lacks an underlying theory for the software engineering process models and methods (SEMAT, 2013). Therefore, we strongly believe that by suggesting the SeDB process model, we have substantially contributed to fill in one small gap in this huge and knowledge and experience demanding software engineering domain.
References


[https://www.google.se/search?q=organization+levels+in+business&espv=2&biw=1280&bih=622&tbm=isch&imgil=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253Bhttp%252523A%25252F%2525252Flecture-notes-forstudents.blogspot.com%25252F2010%25252F04%25252Ftypes-of-information-system.html&source=iu&pf=m&fir=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253B&ved=0CCkQyjdqFQoTCKukxK_jyceCFYeOLAdR_kEdw&ei=UEffVau8MoedsgHH8pO4Bw&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D#imgrc=fOFmAJ-W3hJaZM%3A&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D](https://www.google.se/search?q=organization+levels+in+business&espv=2&biw=1280&bih=622&tbm=isch&imgil=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253Bhttp%252523A%25252F%2525252Flecture-notes-forstudents.blogspot.com%25252F2010%25252F04%25252Ftypes-of-information-system.html&source=iu&pf=m&fir=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253B&ved=0CCkQyjdqFQoTCKukxK_jyceCFYeOLAdR_kEdw&ei=UEffVau8MoedsgHH8pO4Bw&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D#imgrc=fOFmAJ-W3hJaZM%3A&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D)


Management Levels (ud.) *Levels of Organizational Management.*
[https://www.google.se/search?q=organization+levels+in+business&espv=2&biw=1280&bih=622&tbm=isch&imill=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253Bhttp%252523A%25252F%2525252Flecture-notes-forstudents.blogspot.com%25252F2010%25252F04%25252Ftypes-of-information-system.html&source=iu&pf=m&fir=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253B&ved=0CCkQyjdqFQoTCKukxK_jyceCFYeOLAdR_kEdw&ei=UEffVau8MoedsgHH8pO4Bw&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D#imgrc=fOFmAJ-W3hJaZM%3A&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D](https://www.google.se/search?q=organization+levels+in+business&espv=2&biw=1280&bih=622&tbm=isch&imill=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253Bhttp%252523A%25252F%2525252Flecture-notes-forstudents.blogspot.com%25252F2010%25252F04%25252Ftypes-of-information-system.html&source=iu&pf=m&fir=lttnKKE96c8MAM%253A%2523BEIFUPkR1kCeR7M%253B&ved=0CCkQyjdqFQoTCKukxK_jyceCFYeOLAdR_kEdw&ei=UEffVau8MoedsgHH8pO4Bw&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D#imgrc=fOFmAJ-W3hJaZM%3A&usg=__qcB_JluZQQYFHZY_UbgW4ndu1ME%3D)


**Glossary**

**Internet of Things (IoT)**
IoT is the network of physical objects, embedded electronic software or sensors to achieve greater value and services by use of data.

**Legacy System**
A legacy system is a software system that exists in organizations and embodies much of the organization’s processes and knowledge. Therefore, the software holds considerable value to the organization, represents past investments and may not be replaced easily. However, a legacy system may have existed in the organization for years, and is therefore, written in older languages and uses old software engineering technology. Legacy software is, therefore, difficult to modify and maintain.

**Organizational echo system**
A system or a group of interconnected and interacting elements.

**Practices**
A habitual or customary action or way of doing something.

**Morphemes**
The smallest grammatical and meaningful unit in a language.

**Usability**
Capable of being used: in good enough condition to be used.

**Usefulness**
The quality of having utility and especially practical worth or applicability
Appendix A List of Quality Attributes

This appendix lists some of the quality attributes characterizing software systems. Due to the substantial size and diversity of their individual definitions, we only list them in Table A.1. Interested readers are welcome to find the definitions on their own in various publications. We list them herein for providing evidence that defining system quality is not an easy task. Quality can mean different things in different contexts. Therefore, organizations should determine their own quality attributes that suit their systems and contexts.

Table A.1. Examples of Quality Attributes (Berander et al. 2005)

<table>
<thead>
<tr>
<th>Quality Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctness</td>
</tr>
<tr>
<td>Usability</td>
</tr>
<tr>
<td>Testability</td>
</tr>
<tr>
<td>Reusability</td>
</tr>
<tr>
<td>Modifiability</td>
</tr>
<tr>
<td>Understandability</td>
</tr>
<tr>
<td>Generality</td>
</tr>
<tr>
<td>Recoverability</td>
</tr>
<tr>
<td>Change-ability</td>
</tr>
<tr>
<td>Operability</td>
</tr>
<tr>
<td>Traceability</td>
</tr>
<tr>
<td>Access Control</td>
</tr>
<tr>
<td>Execution Efficiency</td>
</tr>
<tr>
<td>Device Independence</td>
</tr>
<tr>
<td>Conciseness</td>
</tr>
</tbody>
</table>
Appendix B

This appendix presents the commissioning companies and their business relationship. It also presents questions that were asked with the purpose of inquiring about Comau and its database requirements.

### B.1 Presentation of Comau Robotics

Comau Robotics is one of the leading suppliers of industrial robots for automotive manufacturing. It is part of the Fiat group. It supplies robots for automotive manufacturing for all major car manufacturers in Europe, America and Asia. Comau is based in Turin, Italy and employs over 14,500 people in 13 different countries. (Comau, 2015)

### B.2 Presentation of CNet

CNet is a company based in Stockholm, Sweden. They employ fifteen people and develop software by participating in various research projects. They have developed software for companies working in content rich industries such as media, construction and facility management and professional publishing. In recent years, the company has been devoted to contributing to the Internet of Things (IoT). One product CNet offers is a sensor recording data about the energy consumption of electronic devices. (CNet, 2015)

### B.3 CNet and Comau

During the last few years, CNet has provided Comau with sensors for their robots in one of their automotive factories. The sensors register data about the robots’ energy consumption only in real time. Today, Comau does not have any long-term solution for the data storage. For this reason, Comau has hired CNet to choose an appropriate solution for registering and viewing data in real-time as well as for storing historic data. It is essential that the data can be viewed in real-time and that the historic data can be quickly accessed. CNet has come to the conclusion that this venture, together with other IoT projects, may require new database technologies. Today, CNet has limited experience in database selection processes and seek guidance on the database market.

Table B.1. Questionnaire used while exploring the information about the commissioning companies and their database requirements

| 1. | What is your role within CNet? |
| 2. | For how long have you had this role? |
| 3. | What other roles did you have before? |
| 4. | Could you please describe Comau? |
| 5. | Do you have any description of the company? |
| 6. | Could you please describe CNet? |
| 7. | Do you have any description of the company? |
| 8. | Could you explain the commissioned work? |
| 9. | What is the purpose of the work? |
| 10. | What are Comau’s requirements for this commissioned work? |
| 11. | How is Comau going to use the work results? |
| 12. | Does CNet have enough competence to solve the commissioned work? |
| 13. | What are CNet’s requirements for this commissioned work? |
Appendix C

This appendix presents the steps presented in the SeDB model. Section C.1 lists and describes the activities in its initial version. Section C.1 lists and describes the activities in its improved version.

C.1 Activities in the Initial Version of the SeDB Process Model

**Activity ID 1: Identify a need for a new database**

All acquisitions, whether they apply to a database or something else, should be based on some need. A need is something useful that is currently missing in an organization. Hence, while making decisions on choosing a database, the companies should ask themselves the question whether they really need it or not.

Establishing a need begins with identifying why a new database is desired or required. Often a need is dependent on an existing problem ranging from problems with the currently used database to both the technological and business environmental problems. Problems with the current database may refer to (1) its inability to efficiently scale a database, or (2) its schema restrictions leading to overly complex data models or (3) its relaxed data consistency and concurrency leading to various restrictions in viewing data. Problems related to the technological environment, on the other hand, may deal with (1) outaged software and hardware platforms or (2) a need to change applications that require a new type of a database. Finally, problems dealing with the business environment may imply that the company needs to select a database to improve its business operation. Otherwise, it may lose customers to its competitors.

**Activity ID 2: Identify a business opportunity for selecting a database**

All types of business are always driven by some business opportunity. There are many definitions of business opportunity (Business Dictionary, ud; Law Dictionary, ud; Martins, ud). In this model, however, we define it as a set of circumstances that makes it beneficial for business to select a new database. The benefits may be both tangible and/or intangible (Reed, ud). Tangible benefits may, for instance, imply lowering labor cost or improving productivity whereas intangible benefits may imply improved personnel morale.

A business opportunity often implies a substantial investment in time and resources. If there is no business opportunity underlying a selection of a new database, then the company may run many risks such as impaired business, increased customer dissatisfaction, or it may even go out of business. Therefore, the decision to acquire a new database must always be based on a clear business opportunity.

There are many examples of business opportunity related to the selection of a database. For instance, in the context of big data, acquiring appropriate databases may help companies store more data about their customers. This, in turn, may help them get a better understanding of their customer needs, and thereby, re-develop their products and become more profitable. Or, the choice of an appropriate database may facilitate update of a content management system (CMS).

**Activity ID 3: Evaluate current business model**

When investing in a new database there is first a need to evaluate the current business model. A business model describes the means of how a company creates, delivers and captures business value. The value then leads to revenue and profit. It encompasses all aspects of the company’s business such as distribution, pricing, advertising, and primarily collecting revenues. The purpose of evaluating the
current business model is to appraise the current business position of the organization.(Chesbrough Rosenbloom, 2002; Osterwalder, 2004)

**Activity ID 4: Evaluate how the current database supports the business model**

After evaluating the overall business model, organizations need to evaluate how the current database supports the business model. There are a number of things that need to be evaluated. These are listed below.

- **Evaluate the use of the current database** - The current database may be used throughout the organization and many applications may rely on the database. Therefore, it may be an integral part of the entire organization. Also, the database may be used during specific periods of time. Therefore, the evaluation also needs to encompass the aspect of time.

- **Evaluate the database understanding** – The database may be old and the documentation may be limited. Organizations need to evaluate whether the database understanding poses problems such as problems with evolving it or educating new employees in the new database technology.

- **Assess the business value of the database** – The business value of the database is an integral part in assessing the current business model in which the current database stores business knowledge and expertise.

- **Determine database quality** – Assessing the quality and its business value is strongly correlated to choice of decision. Databases with low quality and with low business value should be replaced whereas databases with high business value and high quality should be kept. Databases with high business value and/or low business value and high quality may or may not be kept.

- **Check the supplier stability** – This concerns the suppliers of the database. One should evaluate whether the database vendor is going to exist in a predetermined period of time.

- **Evaluate the hardware and software environment** – It is important to evaluate the hardware and software environment of the current database. It may be old and prone to failure, it may imply high maintenance costs, lead to many interoperability problems and it may present performance issues.

**Activity ID 5: Evaluate target business model**

When investing in a new database, one should evaluate the future business model. The goal is to appraise how the business may change with an investment in a new database. The gap between the current and future business models should be fulfilled by the business opportunity.

**Activity ID 6: Identify impact of introducing a new database**

All changes to the technological portfolios lead to various changes. Therefore, the impact of those changes needs to be identified. The following needs to be done:

- **Determine changes to be done** – Investments in a new database may imply various changes that need to be done in other parts of the organization. For example re-writing applications, restructuring the business processes, hiring or firing people, buying new hardware and software and other changes.

- **Assess impact of the endeavor** – The identified changes should be assessed for their impact on the overall software, hardware, organization and business environment.
**Activity ID 7: Estimate cost**

It is important to estimate the cost of selecting a database and implementing it in the company’s environment. Acquiring a new database implies certain overall costs that need to be estimated. Examples of those costs are: (1) the cost of selecting a database, (2) the cost of putting a database into operation, (3) the cost of buying a new technical environment and/or of adapting an old one, (4) the cost of education and training, and (4) other costs that are specific for a particular company and database. All these overall costs must be identified upfront.

Regarding the cost of selecting a database, it refers to all the expenses needed to choose an appropriate database. It covers the study of the database market, study of business opportunity, selection of database candidates, comparison of them and the selection of the most appropriate database.

The selected database does not live an isolated life. When being put into an operation, it is always placed in some technological and operational environment. Hence, while considering the cost of selecting a database, one must always consider the cost of buying a new technical environment and/or adapting the old one to the newly acquired database. This, in turn, implies costs of software, hardware and labor cost for either setting up the environment or to adapting an old one. (Heemstra, 1992)

When choosing a new database, one should always pay heed to the need of educating and training all the people involved (Khan, Kajko-Mattsson 2011). The cost of education and training refers to the expenses associated with educating the workforce in the new technologies. This type of cost should not be underestimated. New technologies may also bring new ways of working and cooperating with the customers which, in turn, may imply costs in form of money and time to educate people and/or time to change their working habits.

**Activity ID 8: Make a feasibility study**

Having the initial cost estimate, companies should conduct a feasibility study. A feasibility study is an assessment of a proposed business venture. Its aim is to assess the prospects of success. In our case, this implies the prospect of succeeding with the acquirement and implementation of a new database and the prospect of achieving the target business model. This is done by uncovering strengths, weaknesses, opportunities and threats of the proposed database acquisition. Here, companies should mainly focus on doing the following

- Analyzing the existing database to find out whether it is worth migrating to a new one. If there is no database, then the company should analyze its operation to find out whether it is worth to acquire one.
- Studying the company’s operation to find out whether it gets impacted by the introduction of a new database.
- Analyzing the environment and all the dependencies between the environment and the new database. Here, by environment, we mean the environmental echo system that the database will be deployed in, hardware and software platforms, applications, people, external partners interacting with the database, and the like.
- Find out whether resources are sufficient for acquiring the database. This includes people resources and their competence, time, and financial resources.

To make a feasibility study is critical because it forces companies to examine all the aspects of the proposal from broader to more narrow perspectives (Lohrey, ud). In this way, it helps the companies to give a proper forethought to the task, and thereby, aids them in assessing the potential of selecting a
new database, in clarifying various uncertainties, and thereby, it helps them understand whether they will succeed with the fulfillment of the business opportunity.

**Activity ID 9: Assess risks**
The feasibility study should always be assisted by risk management. Risk is anything that has not yet happened but may happen, and thereby, jeopardize the success of the project. When investing in a new database, various risks may arise. Examples of such risks are (1) employees disinterest (inertia) in using the new database, (2) architectural complexity that requires specialists within the domain, specialists that may not be easily available, or (3) delay in the delivery of the database, and thereby, the overall delay of the database acquisition project. All risks must be identified, analyzed and planned for.

**Activity ID 10: Assign a budget for selecting a database**
All decisions to proceed with a database acquisition must be followed by a budget assignment. A budget is an estimation of revenue and costs over a future period of time. It must be assigned to all foreseeable expenses. When investing in a new database, there are a number of expenses that need to be accounted for. For example, carrying out educational training about the new database, migrating data from the old database, adapting the applications that rely on the database, documenting the new database and the potential costs for acquiring the software and hardware. A budget is important because it defines the overall spending required by the business venture. It is important to assign a good amount of capital to see the project through. Examples of costs that need to be identified and planned for are:

- Training – Training and educating the employees in the new technology.
- Migration – Migrating data from the old database to the new one.
- Environment – Possible changes to various environments where the database interacts such as, for example, applications that rely on the database.
- Documentation – Documenting the database architecture and documenting all the changes to the database echo system.

**Activity ID 11: Make a decision on whether to proceed with the database selection**
After assessing the risks and conducting a feasibility study, it is time to make a decision. Here, all the results of the previous steps should be taken into account. This includes the current and target business models, feasibility study results, clear and well-motivated business opportunity, and well analyzed risks. The decision may be (1) yes, (2) no, (3) or rework.

**Activity ID 12: Inform the employees about the decision**
When investing in a new database, it is important to inform the employees about the decision. When informing the employees, the companies should present the decision and motivate why it had to be made. They should explain what the organization wishes to accomplish, how the database selection process is going to be performed, how it is going to impact everybody within the organization, what is needed from the employees in order to move forward with the process. This is important in order to minimize or even eliminate the risks associated with employees’ disinterest in or inertia towards the new database.

**Activity ID 13: Elicit requirements guiding the selection of a database.**
All requirements guiding the selection of a new database must be identified. The requirements are functional, nonfunctional, environmental and competence requirements. They are explained below.
Elicit Functional Requirements – Functional requirements focus on what functionalities the database should accomplish and how. They describe the behavior of the database such as storing, accessing, and the like.

Elicit Nonfunctional Requirements – Nonfunctional requirements focus on the non-functional aspects of a database. These may concern product requirements, organizational requirements and external requirements. Product requirements could be usability or reliability concerns of the new database. Organizational requirements could concern certain standards and external requirements could deal with interoperability with applications as well as ethical and legislative issues.

Elicit Environmental Requirements – Environmental requirements are concerned with the organizational echo system in which the database operates. They may concern other databases, applications, hardware, software, operating systems, people, collaborating companies and the like.

Elicit Competence Requirements – Competence requirements are concerned with the competence needed to understand and work with the selected database. Software employees will need to write applications, make logical and physical database design, or they will simply use the new database.

Activity ID 14: Select candidate databases
There are many databases on the market offering a wide range of database schemas that are designed for different purposes. To study them all may be too difficult and time consuming. Hence, the organizations must focus on selecting a subset of them. To identify such a subset requires that one must follow some properties of the databases that are of value for the company. Such properties constitute the selection criteria. The selection criteria should take into account the results of the previous activities. The criteria should be based on the following.

- Criteria related to functional and nonfunctional requirements.
- Criteria related to competence and environmental requirements.
- Other criteria, such as the cost of the database, its support, and the like.

Those criteria should then be transformed to benchmarking criteria, detailed criteria against which the selected databases will be compared.

Activity ID 15: Select databases for testing
Selection of databases is based on the benchmarking criteria, The number of selected databases is dependent on the budget assigned, the potential cost, the time assigned, and the number of good candidates.

Activity ID 16: Test the databases
When databases have been selected, it is high time the organizations tested them. Testing databases implies defining test cases, setting up the test environment and actually testing the databases. Below, we briefly describe those activities:

- **Defining test cases**: Definition of test cases is based on the requirements elicited in the previous steps. The tests should aim at answering the question whether or not the database can perform and handle the requirements. The results of test cases should be carefully examined to ensure that they cover the stated requirements.
- **Test Environment**: Testing environment includes testing of all the necessary hardware and software used to perform the tests.
- **Testing**: Testing can be done on one database at a time or on all the databases in parallel. Irrespective of how it is done, it is important to perform testing on all the tested databases in exactly the same manner and with equal opportunities.
Activity ID 17: Choose a database
Based on all the testing results, the best database is chosen. The testing results should show which of the benchmarking criteria the database could fulfill.

C.2 Activities in the Initial Version of the SeDB Process Model
This section describes the activities in the improved version of the SeDB process model. Due to their substantial amount, we cannot describe them as thoroughly as the activities in the initial model. We only briefly define them.

CS: Determine condition and supportiveness of the currently used database

CS1: Identify/re-identify (a) problem(s) with the current database
This activity identifies problems with the current database within the organization. A problem is a gap between the existing state and the desired state of the organization.

CS2: Identify/re-identify underlying root cause(s) to the problem(s)
This activity is tightly coupled to the problem. Often a perceived problem has an underlying root cause to why the problem is experienced. The root cause can often be some structure of the organization that poses constraints or is managed in the wrong way.

CS3: Determine/re-determine consequences of the problem(s)
This activity determines the consequences and effects of the problem.

CS4: Identify/re-identify a need
This activity is explained in the initial SeDB model (Activity 1).

DBS: Determine that a new database solution is needed
This section describes activities whose common goal is to determine that a new database solution is needed.

DBS1: Evaluate/re-evaluate current business model
This activity is explained in the initial SeDB model (Activity 3 & 5).

DBS2: Evaluate/re-evaluate how the current database supports the current business model
This activity is explained in the initial SeDB model (Activity 4).

DBS3: Determine/re-determine which type of a database is optimal for the business
This activity deals with the identification of the database that will support the organization’s business in an optimal way.

DBS4: Identify/re-identify business opportunity related to the acquisition of a new database (type)
This activity is explained in the initial SeDB model (Activity 2).

DBS5: Define/re-define future business model
This activity is explained in the initial SeDB model (Activity 6).

BV: Determine/re-determine business value
This activity is explained in Activity ID 3 with the business opportunity and its tangible and intangible benefits.
**DBS6: Confirm/re-confirm that a new database solution is needed**
This activity is a confirmation that a new database solution is needed.

**MS: Manage stakeholders**
A stakeholder is a person or a group of people or organization that that has interest or concern in an organization.

**MS1: Identify/re-identify stakeholders experiencing the need**
This activity identifies the stakeholders that are experiencing the current need for a new database solution.

**MS2: Identify/re-identify stakeholders experiencing the problem**
This activity identifies the stakeholders that are experiencing the problem with the database solution.

**MS3: Identify/re-identify stakeholders that will use the system**
This activity identifies the stakeholders that will use the new database solution.

**MS4: Identify/re-identify stakeholder that will be involved in selecting a database**
This activity identifies the stakeholders that will actively participate in the process of selecting the new database.

**MS5: Identify/re-identify stakeholder that will fund the new database and its implementation**
This activity identifies the stakeholders that will fund the process of selecting the new database.

**DI: Identify/re-identify stakeholder to be informed about the database selection process**
This activity identifies the stakeholders that will be informed about the progress of the database selection process.

**DI: Disseminate information about the new database selection endeavor**
These activities are concerned with informing various people involved in the selection process.

**DI1: Determine communication channels to be used for information dissemination**
This activity identifies how to communicate information about the progress of the database selection process.

**DI2: Determine milestones for information dissemination**
This activity is concerned with determining what milestones are to be used when disseminating information within the organization.

**DI3: Disseminate information, when relevant**
Not always, one may plan in advance when to disseminate information. This activity is concerned with disseminating information if sudden changes or unplanned activities present themselves.

**D: Make a decision**
This section lists the activities that must be performed for making educated decision on selecting a new database.
D1: Study all the data underlying the decision making
It is important that management has knowledge and understanding of the problems and needs leading to the database selection process. Also, the management should have good control over the process in order to make right decisions.

D2: Study the current and target business models
This activity is explained in the initial SeDB model (Activity 3 and 5).

D3: Study the suggested optimal database solution
This activity is concerned with studying the optimal database solution in order to make a decision.

D4: Study alternative database solutions
This activity is concerned with studying the alternative database solutions in order to make a decision.

D5: Study the cost of choosing the optimal versus the alternative solution
This activity is concerned with studying the cost for choosing the optimal versus alternative solutions. In some cases, the optimal solution can be very expensive and not worth the decision of moving forward. The alternative solutions may solve most of the needs and problems and cost less than the optimal solution.

D6: Study the impact of choosing the optimal versus the alternative solution
This activity is concerned with studying the impact of choosing the optimal solution versus the alternative solution(s).

D7: Study the business value
This activity is explained in the initial SeDB model (Activity 2).

D8: Make a decision
This activity is concerned with the decision on what database to choose or the decision to acquire a new database (see Activities 8.1 and 8.2).

D8.1: Make a decision whether to proceed with the database selection process
This activity deals with the decision on whether to move forward with the selection process.

D-D-8.2 Make a decision whether to acquire the database
This activity is concerned with the decision on whether the winning database should be chosen.

P: Plan
Planning aims at outlining a plan of the available resources in order to achieve optimum results. It consists of identifying goals or milestones, formulating strategies, identifying constraints and monitoring its progress.

P1: Define milestones and deliverables
A milestone is a sub-objective created to optimize the planning and monitoring of a project. Deliverables are items or reports that need to be completed. This activity aims at defining milestones and deliverables to achieve optimal results.

P2: Identify project constraints
A constraint is something viewed upon as a bottleneck. It restricts the overall progress of the mission, and therefore, it hinders it from achieving its full potential or value. This activity is concerned with identifying these constraints in order to be able to counteract them in the best possible way.
P3: Define/re-define project schedule
A schedule is a timetable for the project. It includes milestones and deliverables and it shows how the project proceeds over the course of the timetable. This activity is concerned with plotting the identified milestones and deliverables in a schedule in order to aid Activity P4 dealing with monitoring the progress.

P4: Monitor progress against the plan
Monitoring the progress is concerned with the activity of supervising the progress against the defined plan in Activities P1, P2 and P3. It is concerned with meeting the milestones and deliverables and the overall target of the project.

SDB: Select a database
This group includes the activities dealing with the actual selection of databases.

SDB1: Elicit the requirements guiding the selection of the new database
Eliciting requirements is the way of collecting the desired demand on the system. Requirements are typically elicited from users, customers and stakeholders. This activity is concerned with gathering and identifying the necessary requirements that the database must fulfill.

SDB2: Assure usefulness of the requirements
Not all requirements are useful for a database selection process. Hence, it is important that each requirement be evaluated from its usefulness perspective.

SDB3: Define/re-define test cases
Test cases are a set of conditions to ensure that the database works as expected. Test cases are based on the elicited requirements and aim at answering whether the database fulfills these requirements.

SDB4: Determine type of a database needed to fulfill the solution
There are over one million databases to choose from on today’s market. A database type is a general category of databases that has a similar structure. Database types are presented in Chapter 3.1. Determining what type of database that is best suited to fulfill the solution is not easy. However, this activity aims at selecting a sub-set of the databases based on the elicited requirements as well as the test cases.

SDB5: Select candidate database to be studied
When choosing candidate database to be studied, the requirements elicited as well as the test cases need to be studied in order to make a decision. The candidate databases should be part of the initial sub-set of databases that were selected in Activity SDB4. However, this activity involves the actual products and not just the type of a database.

SDB6: Test the candidate databases
This activity aims at testing the candidate databases that were selected in Activity SDB5 with the test cases defined in Activity SDB3.

SDB7: Rank the candidate databases using the testing results
After the tests have been completed, it is important to rank the candidate databases. The ranking should mainly be based on their performance. Some database may not have fulfilled the necessary requirements.
**SDB8: Suggest the optimal database solution that is appropriate for the company**  
Suggestion of the optimal database solution is based on the testing results gathered during Activity SDB7.

**SDB9: Suggest alternative database solutions that are appropriate for the company**  
Suggestion of the alternative database solutions is based on the testing results gathered during Activity SDB7.

**BV: Determine business value**  
This activity is explained in Activity ID 2 together with the business opportunity and its tangible and intangible benefits. However, this activity goes into more detail regarding the business value.

**BV1: Define/re-define impact of introducing a new database (type)**  
Defining the impact of introducing a new database is concerned with identifying the benefits of introducing a new database. Here, one should define what type of tangible and intangible benefits it may have. Impact is also related to what negative effects an introduction of a new database may imply.

**BV2: Define/redefine impact on the stakeholders**  
Defining impact on the stakeholder is concerned with identifying what stakeholders may be impacted by a new database.

**BV3: Define/redefine impact on the organizational echo system**  
The introduction of a new database may imply an entire organizational makeover. Many applications relying on the database have to be rewritten and changed. New hardware and software platforms have to be acquired. Hence, the impact on the organizational echo system should be addressed.

**BV4: Define/redefine impact on the business**  
An introduction of a new database may impact the business by the unbeknowing of certain events. Previous activities such as BV3 may show that the introduction of a new database implies large changes to the organizational echo system. Hence, it is important to define what impact this has on the business.

**BV5: Estimate/re-estimate the cost**  
This activity is explained in the initial SeDB model (Activity 7).

**BV5.1 Estimate/re-estimate the cost of selecting a database**  
This activity is explained in the initial SeDB model (Activity 7).

**BV5.2 Estimate/re-estimate the cost of implementing the database**  
This activity is explained in the initial SeDB model (Activity 7).

**BV5.3 Estimate/re-estimate the cost of evolving and maintain the database**  
This activity is explained in the initial SeDB model (Activity 7).

**BV5.4 Estimate/re-estimate the cost of operating the database**  
This activity is explained in the initial SeDB model (Activity 7).

**BV5.5: Estimate/re-estimate the cost of changing the organizational echo system**  
This activity is explained in the initial SeDB model (Activity 7).
BV5.6 Estimate/re-estimate the cost of educating and training all the stakeholders involved
This activity is explained in the initial SeDB model (Activity 7).

BV5.7: Estimate/re-estimate other costs that are specific for the company and business
This activity is explained in the initial SeDB model (Activity 7).

BV6: Determine success criteria related to the acquisition of the new database
Success criteria are the combination of certain facts that are required to accomplish a successful database selection project.

BV7: Determine outcome of the database acquisition process
Data selection processes may result in different outcomes. Often, they are strongly related to the business value, success criteria, company’s budget and/or its willingness to spend money on the acquisition and similar. Outcome may also imply time to be spent on acquiring a new database. Often, the choice of a new database may imply that the major parts of the echo system will have to be changed. In this case, the outcome may mean company’s willingness of changing the echo system.

BV8: Estimate/re-estimate business value
This activity is explained in Activity ID 3 with the business opportunity and its tangible and intangible benefits.

B: Assign/reassign budget
A budget is an estimation of revenue and costs over a future period of time. It must be assigned to all foreseeable expenses.

B1: Determine budget for the overall database lifecycle
This activity is explained in the initial SeDB model (Activity 10).

B2: Determine budget for the database selection phase
This activity is explained in the initial SeDB model (Activity 10).

B3: Reserve budget for the database lifecycle
This activity is explained in the initial SeDB model (Activity 10).

B4: Reserve budget for the database selection phase
This activity is explained in the initial SeDB model (Activity 10).

B5: Assign budget for the next coming steps of the database selection phase
This activity is explained in the initial SeDB model (Activity 10).

R: Manage risks

R1: Identify/re-identify risks
This activity is explained in the initial SeDB model (Activity 9).

R2: Analyze/re-analyze risks
This activity is explained in the initial SeDB model (Activity 9).

R3: plan/re-plan risk management
This activity is explained in the initial SeDB model (Activity 9).
R4: Monitor and control risks
Monitoring and control of risks is concerned with supervising and making sure that the risks do not jeopardize the project outcome.

R5: Remove risks from the risk list
A risk list is the list of identified risks during the Activity R1. When these risks do no longer threaten the project, they are no longer in need of monitoring and control. Hence, they should be removed from the list.

ET: Educate and train

ET1: Identify stakeholder groups to be educated and trained (Software professionals, users, managers)
A stakeholder is a person or a group that has interest and concern in an organization. To make them contribute to the company’s business, they should be educated and trained in how to use or work with the new database.

ET2: Determine educational needs for each stakeholder group
Certain stakeholders may present different needs in terms of education. Hence, the education and training should be adapted to individual stakeholder needs. Users may need a certain type of education while software professionals assigned to maintain the system need other types of education.

ET3: Educate and train
This activity concerns the process of educating and training the stakeholders as identified in the Activities ET1 and ET2.
Appendix D Matching Terminology of Validity Threats within Qualitative and Qualitative Research

This appendix demonstrates the validity threats in both quantitative and qualitative research. They are presented in Table D.1.

Table D.1. List of validity threats in quantitative and qualitative research

<table>
<thead>
<tr>
<th>Quantitative Research</th>
<th>Qualitative Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Validity</td>
<td>Credibility</td>
</tr>
<tr>
<td>External Validity</td>
<td>Transferability</td>
</tr>
<tr>
<td>Reliability</td>
<td>Dependability</td>
</tr>
<tr>
<td>Objectivity</td>
<td>Conformability</td>
</tr>
</tbody>
</table>
Appendix E Test Cases and Testing Results

In this appendix, we first present the test cases used in the benchmarking process and the testing results. Section E.1 describes the test cases whereas Section E.2 presents the testing results.

E.1 Test Cases

In this section, we present the test cases for the benchmarking requirements SR1, SR2, SR3 and SR4 that were used during the testing process. Our descriptions are supported by Figures E.1 – E.3.

Test cases dealing with the ability to store x number of data metrics for x number of robots during one time unit (fulfilling the requirement SR1)

Test cases 1-3 concerned the requirement SR1 and aimed at evaluating the database ability to store a required amount of data records from 1 to 100 robots. The test cases are presented in the grey box on the top left hand side of Figure E.2. One manufacturing process may consist of up to one hundred robots. All of the values that these robots produce need to be stored in the database. The input from the robots are the test data P1 to PF3 together with the robot ID and timestamp. Here, the timestamp is the key and we use it to aggregate a number of values stored during one second.

Test case 1 tests the ability to store data gathered from one robot during one second. Bearing in mind the fact that one robot produces data every 200 milliseconds, the output should then be five data records. Test case 2 tests the ability to store data gathered from ten robots during one second. The output should then be fifty data records. Test case 3 tests the ability to store data gathered from one hundred robots during one second. The output should then be five hundred data records. The outputs of test cases 1, 2 and 3 are presented in Figure E.1.

Test cases dealing with the ability to perform aggregation with x number of metric values by one robot per one time unit.

Test cases 4-51 concerned the requirement SR2 and aimed at evaluating the databases ability to perform sums and averages of the stored data. They are presented in the white part of Figure E.1. Comau has requested that the database should deal with sums and averages of data records over certain time periods. The ability to aggregate values gives evidence of the ability to view the data in a different way and may lead to various data analyses. Tests 4-15 and 28-39 test the database ability to
| Test ID: 1 | Test name: **The database stores five data metrics for one robot per second.**  
Input: Test data including values \(P_1, P_2, P_3, Q_1, Q_2, Q_3, S_1, S_2, S_3, PF_1, PF_2, PF_3\) gathered from one Comau robot during one second.  
Output: Evidence of five data insertions per robot per second. |
| Test ID: 2 | Test name: **The database stores fifty data metrics for ten robots per second.**  
Input: Test data including values \(P_1, P_2, P_3, Q_1, Q_2, Q_3, S_1, S_2, S_3, PF_1, PF_2, PF_3\) gathered from ten Comau robots during one second.  
Output: Evidence of fifty data insertions per robot per second. |
| Test ID: 3 | Test name: **The database stores five hundred data metrics for 100 robots per second.**  
Input: Test data including values \(P_1, P_2, P_3, Q_1, Q_2, Q_3, S_1, S_2, S_3, PF_1, PF_2, PF_3\) gathered from 100 Comau robots during one second.  
Output: Evidence of five hundred data insertions per robot per second. |
| Test ID: 4-15 | Test name: **The database stores the sum of individual value \(P_1, P_2\) or \(P_3\) or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) collected from one robot during one second.**  
Input: Sum gathered from a Comau robot during one second. Only the individual robot value \(P_1\) or \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) is considered.  
Output: Sum of the individual robot values \(P_1, P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) from a robot during one second. |
| Test ID: 16-27 | Test name: **The database stores the sum of individual value \(P_1, P_2\) or \(P_3\) or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) collected from one robot during ten seconds.**  
Input: Sum gathered from a Comau robot during ten seconds. Only the individual robot value \(P_1\) or \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) is considered.  
Output: Sum of the individual robot values \(P_1, P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) from a robot during ten seconds. |
| Test ID: 28-39 | Test name: **The database stores the average of individual value \(P_1, P_2\) or \(P_3\) or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) collected from one robot during one second.**  
Input: Average gathered from a Comau robot during one second. Only the individual robot value \(P_1\) or \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) is considered.  
Output: Average value of the individual robot values \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) from a robot during one second. |
| Test ID: 40-51 | Test name: **The database stores the average of individual value \(P_1, P_2\) or \(P_3\) or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) collected from one robot during ten seconds.**  
Input: Average gathered from a Comau robot during ten seconds. Only the individual robot value \(P_1\) or \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) is considered.  
Output: Average value of the individual robot values \(P_2\) or \(P_3\), or \(Q_1\) or \(Q_2\) or \(Q_3\) or \(S_1\) or \(S_2\) or \(S_3\) or \(PF_1\) or \(PF_2\) or \(PF_3\) from a robot during ten seconds. |

Figure E.2. Test cases fulfilling Requirements SR1-SR2

count and present sums and averages of the values from \(P_1\) to \(PF_3\) for one robot during one second, respectively. Tests 16-27 and 40-51, on the other hand, test the database ability to count and present sums and averages of the values \(P_1\) to \(PF_3\) for one robot during ten seconds, respectively. The outputs of these test cases should be sums and averages for each data value. Examples of the output of test cases 4 and 28 are presented in Figure E.1.
<table>
<thead>
<tr>
<th>Test ID: S2-63</th>
<th>Test name: The database continuously stores and presents individual robot value P1 or P2 or P3 or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3 collected from one robot with millisecond precision.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>Test data continuously gathered from one Comau robot. Only individual value P1 or P2 or P3, or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3 is considered.</td>
</tr>
<tr>
<td>Output:</td>
<td>Evidence of storing and representing robot values (P1 or P2 or P3 or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3) from a robot with a millisecond precision.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test ID: 76-87</th>
<th>Test name: The database continuously stores and presents the average of individual robot value P1 or P2 or P3 or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3 collected from one robot with one second precision.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>The average continuously gathered from one Comau robot. Only individual value P1 or P2 or P3, or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3 is considered.</td>
</tr>
<tr>
<td>Output:</td>
<td>Evidence of storing and representing average robot values (P1 or P2 or P3 or Q1 or Q2 or Q3 or S1 or S2 or S3 or PF1 or PF2 or PF3) from a robot with one second precision.</td>
</tr>
</tbody>
</table>

Test cases fulfilling Requirements SR3-SR4

** Test cases dealing with the ability to continuously store and present individual robot values collected from one robot per time unit without any delay. **
Tests S2-63 concerned the requirement SR3 and aimed at evaluating the database ability to continuously store and present individual data values during the course of a certain time period. The test cases are presented in the grey part of Figure E.3. The representative output from these tests is presented in Figure E.1.

** Test cases dealing with storing the sums and averages of individual values as collected by one robot during one time unit. **
Test cases 64-87 concerned the requirement SR4 and aimed at evaluating the databases ability to aggregate sums and averages of the robot data. These tests are presented in the white part of Figure 7.6. Comau has requested that the database should be able to present the aggregates of data records over certain time periods. This may present the ability to view the data in a different way and may lead to various data analyses. Test case 64 tests the ability to display the sum of values P1 to PF3 for one robot during one second. Test case 76 tests the ability to display the average of the values P1 to PF3 for one robot during ten seconds. The output of these test cases should be the average value for each data value during that time instance. Example of such output is presented in Figure E.1.
7.2.5.3. Setting up Testing Environment

The two databases tested, MongoDB and OpenTSDB, required two different testing environments. MongoDB’s testing environment consisted of Visual Studio, C#, MongoDB C# driver, MongoDB and a Laptop running Windows OS. When testing OpenTSDB, an additional testing environment was setup consisting of three node HDInsight Hadoop and HBase cluster, a virtual machine with OpenTSDB, Visual Studio, C# and a laptop running Windows OS.

E.2 Testing Results

Two databases, OpenTSDB and MongoDB were tested against 86 test cases as described in Section 7.2.5.1. Table E.1 presents the testing results. All the test cases were fulfilled by the two databases except for test cases 52-63 which could not be fulfilled by OpenTSDB.

Table E.1. OpenTSDB and MongoDB Test results.

<table>
<thead>
<tr>
<th>TestID</th>
<th>OpenTSDB</th>
<th>MongoDB</th>
<th>TestID</th>
<th>OpenTSDB</th>
<th>MongoDB</th>
<th>TestID</th>
<th>OpenTSDB</th>
<th>MongoDB</th>
<th>TestID</th>
<th>OpenTSDB</th>
<th>MongoDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>30</td>
<td>✓</td>
<td>✓</td>
<td>59</td>
<td>-</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td>✓</td>
<td>31</td>
<td>✓</td>
<td>✓</td>
<td>60</td>
<td>-</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>32</td>
<td>✓</td>
<td>✓</td>
<td>61</td>
<td>-</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>✓</td>
<td>✓</td>
<td>33</td>
<td>✓</td>
<td>✓</td>
<td>62</td>
<td>-</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>34</td>
<td>✓</td>
<td>✓</td>
<td>63</td>
<td>-</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>✓</td>
<td>✓</td>
<td>35</td>
<td>✓</td>
<td>✓</td>
<td>64</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>✓</td>
<td>✓</td>
<td>36</td>
<td>✓</td>
<td>✓</td>
<td>65</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>✓</td>
<td>✓</td>
<td>37</td>
<td>✓</td>
<td>✓</td>
<td>66</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>✓</td>
<td>✓</td>
<td>38</td>
<td>✓</td>
<td>✓</td>
<td>67</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>✓</td>
<td>✓</td>
<td>39</td>
<td>✓</td>
<td>✓</td>
<td>68</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>✓</td>
<td>✓</td>
<td>40</td>
<td>✓</td>
<td>✓</td>
<td>69</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>✓</td>
<td>✓</td>
<td>41</td>
<td>✓</td>
<td>✓</td>
<td>70</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>✓</td>
<td>✓</td>
<td>42</td>
<td>✓</td>
<td>✓</td>
<td>71</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>✓</td>
<td>✓</td>
<td>43</td>
<td>✓</td>
<td>✓</td>
<td>72</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>✓</td>
<td>✓</td>
<td>44</td>
<td>✓</td>
<td>✓</td>
<td>73</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>✓</td>
<td>✓</td>
<td>45</td>
<td>✓</td>
<td>✓</td>
<td>74</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>✓</td>
<td>✓</td>
<td>46</td>
<td>✓</td>
<td>✓</td>
<td>75</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>✓</td>
<td>✓</td>
<td>47</td>
<td>✓</td>
<td>✓</td>
<td>76</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>✓</td>
<td>✓</td>
<td>48</td>
<td>✓</td>
<td>✓</td>
<td>77</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>✓</td>
<td>✓</td>
<td>49</td>
<td>✓</td>
<td>✓</td>
<td>78</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>✓</td>
<td>✓</td>
<td>50</td>
<td>✓</td>
<td>✓</td>
<td>79</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>✓</td>
<td>✓</td>
<td>51</td>
<td>✓</td>
<td>✓</td>
<td>80</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>✓</td>
<td>✓</td>
<td>52</td>
<td>-</td>
<td>✓</td>
<td>80</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>✓</td>
<td>✓</td>
<td>53</td>
<td>-</td>
<td>✓</td>
<td>81</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>✓</td>
<td>✓</td>
<td>54</td>
<td>-</td>
<td>✓</td>
<td>82</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>✓</td>
<td>✓</td>
<td>55</td>
<td>-</td>
<td>✓</td>
<td>83</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>✓</td>
<td>✓</td>
<td>56</td>
<td>-</td>
<td>✓</td>
<td>84</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>✓</td>
<td>✓</td>
<td>57</td>
<td>-</td>
<td>✓</td>
<td>85</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>✓</td>
<td>✓</td>
<td>58</td>
<td>-</td>
<td>✓</td>
<td>86</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>