LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

How to comprehend, rationalize and develop version control systems? Case Aalto University

Nousiainen, Jani

2016 Laurea

Laurea-ammattikorkeakoulu

How to comprehend, rationalize and develop version control systems? Case Aalto University

Jani Nousiainen
Information Systems
Master's Thesis
December, 2016

Laurea University of Applied Sciences
Degree Programme in Information Systems
Master's Thesis

Abstract

Jani Nousiainen

How to comprehend, rationalize and develop version control systems? Case Aalto University

Year 2016 Pages 63

The topic of the thesis is topical, as the various universities and non-governmental companies are looking for ways to improve their processes to gain budget savings. This thesis will debate on the development, comprehension and rationalization of version control system usage at Aalto University. The research method of the study was case study which consisted of several (n=3) smaller case studies.

The thesis has been conducted during a VCS project, which aimed to clarify the usage of VCS at Aalto University by providing a centrally managed system instead of multiple systems which are currently used among the departments of Aalto University. Objective of the thesis is to examine the possibility to substitute current systems to a one, centrally managed system and clarify the steps important to a VCS acquisition project and suggest improvements to the process of the project.

The topic of the thesis is topical, as the various universities and non-governmental companies are looking for ways to improve their processes to gain budget savings. By enhancing the processes, the project should advance efficiently and the lifecycle of the project from the begin of the project to the finished project can be reduced consequently offering savings to the companies in both, the staff costs and in equipment costs.

The most important result of the research is that the reduction of overlapping systems will significantly improve the usage of the system and the local support provided for the system. At the same time, reduction will also offer a possibility to focus on the development of systems instead of just focusing on administrating multiple systems.

The project itself was conducted at the premises of Aalto University from 2015 to 2016. During that time, three separate studies were conducted which all aimed to provide an answer to the main theme of this study "How to comprehend, rationalize and develop version control systems?"

As the answer to the main theme was answered with the results of three smaller case studies, making the design of this research a multiple-case design. Using a multiple-case design, prevented the critique of simply relying on a one source of information. This critique has been debated by Dubé & Paré (Dubé & Paré, 2003, p. 609) in their study, where they say that using only one case, prevents the results from being generalizable.

Keywords: Version control system, Project, Project management, Rationalization

Laurea-ammattikorkeakoulu **Tiivistelmä**
Leppävaara
Tietojärjestelmäosaamisen koulutusohjelma

Jani Nousiainen

Kuinka ymmärtää, kehittää ja järkeistää versionhallintajärjestelmiä? Tapaus Aalto-yliopisto

Vuosi 2016 Sivumäärä 63

Jatkuvien säästöjen etsimisen aikana yritykset pyrkivät prosessien tehostamiseen. Tämän tutkimuksen tavoitteena on tutkia sitä, miten versionhallintajärjestelmän käyttöä voidaan kehittää, ymmärtää ja järkeistää Aalto-yliopistossa. Tutkimus toteutettiin tapaustutkimuksena, joka koostui useammasta (n=3) pienemmästä tapaustutkimuksesta. Tutkimus suoritettiin Versionhallintajärjestelmän hankintaprojektin yhteydessä. Hankintaprojektin tavoitteena oli tarjota keskitetyssä ylläpidossa oleva järjestelmä, jolla korvattaisiin järjestelmät, joita monet laitoksista tällä hetkellä ylläpitävät itse.

Tutkimuksen tavoitteena on selvittää nykyisten järjestelmien korvaaminen keskitetyllä järjestelmällä ja samalla selkeyttää ja tarjota parannusehdotuksia vaiheisiin, jotka edelsivät varsinaisen projektin aloittamista.

Aihe on hyvin ajankohtainen, sillä monet yliopistot ja yksityisen sektorin yritykset etsivät keinoja tehostaa prosessejaan saavuttaakseen säästöjä sekä henkilöstö että laitteistokuluissa. Tehostamalla prosesseja projekti etenee tehokkaasti sen alusta loppuun ja projektin elinkaaren pituutta voidaan täten lyhentää samalla tarjoten säästöjä yrityksille.

Tutkimuksen tärkein tulos on se, että päällekkäisten järjestelmien vähentäminen tehostaa sekä järjestelmien käyttöä, että paikallista tukea järjestelmälle. Päällekkäisten järjestelmien vähentäminen mahdollistaa myös jäljelle jäävien järjestelmien tehokkaamman kehittämisen.

Tutkimus suoritettiin Aalto Yliopistossa vuosien 2015 ja 2016 aikana. Tänä aikana projektin yhteydessä suoritettiin kolme erillistä tutkimusta, joiden tavoitteena oli vastata koko tutkimuksen yhteiseen teemaan ”Kuinka versionhallintajärjestelmien käyttöä voidaan ymmärtää, järkeistää ja kehittää?”

Vastaus tutkimuksen pääteemaan saatiin kolmen pienemmän tutkimuksen tuloksena. Koska tutkimuksia oli kolme kappaletta, voidaan sanoa, että koko tutkimus suoritettiin monitapaustutkimuksena. Monitapaustutkimus-metodia hyödyntämällä vältettiin tapaustutkimuksiin usein kohdistuva kritiikki siitä, että tulokset nojautuvat vain yhden tietolähteen varaan. Myös Dubé & Paré (Dubé & Paré, 2003, p. 609) keskustelevat tapaustutkimusten kritiikistä tutkimuksessaan sanoen, että vain yhden tapauksen käyttäminen estää tulosten yleistämisen.

Asiasanat: Versionhallinta järjestelmä, Projekti, Projektinhallinta, Järkeistäminen

Table of contents

Abbreviations

CS = Case Study Research

CVCS = Centralized version control system

DVCS = Distributed Version Control

Git = Version control system software

GitLab = Version control system software

HAKA = The identity federation of the Finnish universities, polytechnics and research institutions

HIIT = Helsinki Institute for Information Technology

IT = Information Technology

LDAP = Lightweight Directory Access Protocol

SGPP = Steering Group of Projects and Services of Aalto IT

SVN = Apache Subversion, version controls system software

VCS=Version Control System

1 Introduction

The subject of this research is the development, comprehension and rationalization of the usage of version control systems in Aalto University. The objective of the research is to identify, how the reduction of overlapping version control systems will affect on the usage, development and support of the remaining systems.

The departments at Aalto University currently host more than 10 different version control systems. Hosting and administrating these systems takes out time from development and user support. The aim of the project is to offer one centrally managed version control system for the staff, students and other personnel at Aalto University. Time consumed administrating multiple systems can be routed to the development of a centrally managed system, and at the same time, costs for maintaining several systems will be reduced as the staff costs and expenditure on equipment will also descend.

This study is closely related to information technology; it has plenty of information technology abbreviations. These abbreviations have been reviewed before the introduction section. The single most important result of the research is that the reduction of overlapping systems will significantly improve the usage of the system and the local support provided for the system. At the same time, reduction will also offer a possibility to focus on the development of systems instead of just focusing on administrating multiple systems.

1.1 Operational environment and context of study

Aalto University was established in 2010 when 3 Finnish universities (The Helsinki School of Economics, Helsinki University of Technology and the University of Art and Design Helsinki) were merged together. Currently, the university consists of six schools and two other units. These schools each have a small number of IT personnel of their own. The schools have local IT systems, network environments and access to a network environment that covers the entire university, Aalto-network.

The university also has a centralized IT department, which offers its services for the whole university. These services are such as network services, software and IT purchases to mention but a few. The IT department is responsible for the Aalto-network.

The IT department consists roughly of 120 employees, who are divided into different units, that are all focused on different areas of information technology. These units' upkeep and maintain the current systems and develop and acquire new systems. The IT department also provides IT support in the form of IT service desk. The service desk consists of 7 units, which are located

on different campuses and buildings on campuses. The rest of the department works in one building.
The workstations supported by the IT department and that are connected to the Aalto-network have Linux, OSX or Windows operating systems and the department has acquired licenses for a large group of most commonly used programs. The IT Department will also install Self-acquired software case-by-case.

Version control systems are used widely in the university and the main user group consists of students and researchers. The main use for version control among the students is to keep track of the software code, which they develop for the courses held at university. The most used version control software among students is Git. The second largest user group at the university is researchers and research groups. Researchers either use a version control system provided by their department or a web-based service. Research groups use both. A third group actively using version control systems are the administrators at the university. Administrators store puppet repositories in the GitLab version control system and use it to share settings and programs to the department's Linux environment.

Currently, all the version control environments are administrated by different departments. One department may host more than one version control environment depending on what the staff has required from the local administrators. The usage of the different systems in the 2015 is shown in the Figure 1.

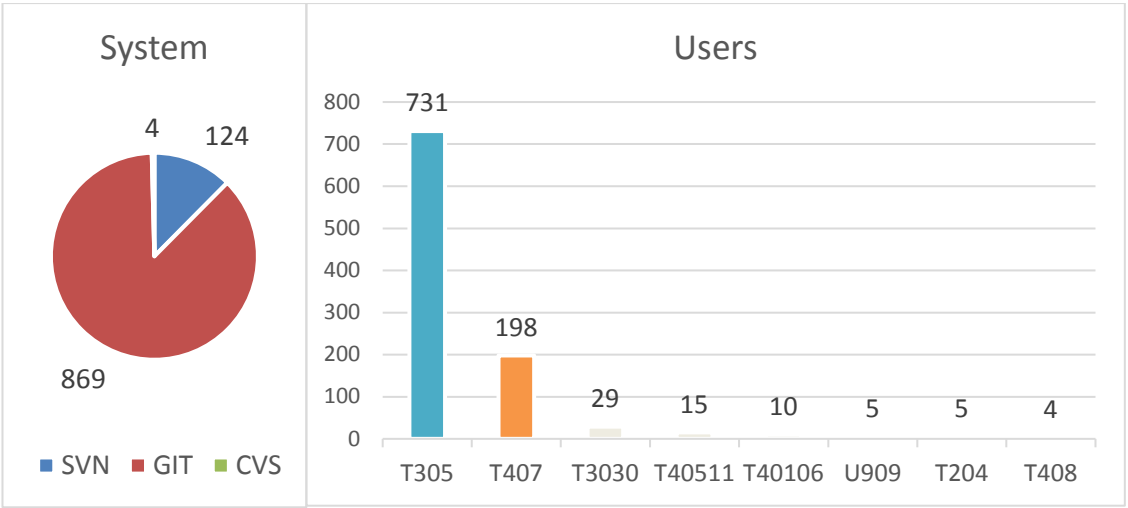

Figure 1 : Different system usage based on interviews and preliminary report (Lähteenmäki, 2015)

Figure 1 shows the usage of current systems in Aalto University. The figure clearly shows that version control systems based on GIT-technology are by far the most popular among the users

of the university. Almost 87% of the users using VCS's are using GIT, 12% are using SVN and 1% are still using CVS which was developed in the mid 80's.

The current situation is that the university is trying to reduce the number of the overlapping systems. Sometimes this can be done simply by shutting down some of them and other times the cur-rent systems need to be replaced with a new and possibly better system before the old one can be shut down.

Aalto University has very little or no written information about some of its processes. There are documentations and employees also have experience and information, but there are risks that this information could vanish. The university is constantly forced to reduce the number of staff due to the ever-decreasing funding and thus make work assignments and processes more efficient. This might lead in the future to a situation where some of the core employees have left the university or have forgotten the information. In this case, the study will become useful. It will provide written information, which can be used in later projects.

1.2 Theme and setting of the study

The subject of the study is "How to comprehend, rationalize and develop version control systems?" and the study takes place in Aalto University's Otaniemi campus between spring 2015 and fall 2016. The study focuses on the current version control systems administrated at different schools of the university and the new VCS that the centralized IT department of the university will purchase and administrate. The study aims to define the acquisition process of the new system, so that the results of this study can be used in future acquisition projects. The study explores the situation of the systems before the acquisition and various parts of the acquisition project.

The theme of the study is to build a general view of the VCS acquisition process, so that findings can be used as a guideline in future projects. Realization of this will be constructed with both theory and by participating in the acquisition process. The research methodologies, research questions, and analysis methods are described in detail later in this study. The methodological approach to this study is case study research (CS). "A qualitative case study examines a phenomenon within its real-life context. Data are collected on or about a single individual, group, or event." (Guest, Namey, & Mitchell, 2012, p. 14.) A series of case studies were conducted to obtain knowledge to answer the research question of this study. These case studies are described in Chapter 3 of the study.

1.3 Research environment and reasoning

The study was compiled prior and during an ongoing VCS project at Aalto University. The research was a qualitative study as the people and the system under study were studied in the context where they normally act (Kaplan & Maxwell, 2005, p. 30; Runeson & Höst, 2009, p.131).

As already mentioned, several different version control systems are maintained at the premises of Aalto University, and the reasoning for the VCS project is to reduce the amount of overlapping systems. By reducing the number of systems administrated, the contribution of local administrators can be focused on more relevant systems.

The primary goal for this project is to create a centralized version control system for the whole university. The secondary goal for this project, and the primary goal for this research is to provide knowledge how the overlapping systems can be reduced and what kind of steps can be taken prior and during the project to have a successful project. The aim of the research is to provide such information and methods, that they can be replicated in similar projects at the university.

1.4 Expected results and future utilization

The results of this study will include data about the tasks which should be done prior, during and after a version control acquisition project. Research will gather data about the composition of the project group and what kind of knowledge is required by the members of the project in different phases of the project to get the full gain of the people participating in the project. The results of this study are expected to give beneficial information about project management. How the project group can be formed successfully, what are the necessary and beneficial tasks that should be done before the actual project, what are the necessary tasks during the actual VCS acquisition project and how they should be done to complete the project successfully. What kind of unexpected changes can there be during a project and how the project should handle them?

The results of this research should be usable during similar projects within the premises of Aalto University. On the other hand, even though the study has been composed at the university environment, there is no reason, why the results could not be used outside the university. The same methods can be adapted to another universities or the corporate life.

1.5 Literature review

The study leaned heavily on literature and theories presented in it. Methodologies used in this study were based on the literature about each methodological procedure. Literature was used to gain knowledge about the processes and how to analyze the data received from the case

studies. The two most used literature subjects during the studies were literature about the methodologies and literature about version control systems and their use.

The most used subject for literature was literature regarding different methodologies and the use of them for research purpose. The whole study was a Case study methodology was adapted from Yin's book, Case study research: Design and methods (Yin, 2009). The book presents case study methodology and means for a successful study.

Due to the qualitative nature of the study, the sage encyclopedia of qualitative research methods (Given, 2008), was used as a guidebook. The book elaborates information, techniques, and examples about the qualitative study research. Since the study was about information systems (IS), Design Research in Information Systems by Hevner and Chatterjee. This book is about Design Science Research (DSR) and covers such topics as people and design, the past and present of software designs, evaluation methods and focus-group use (Hevner & Chatterjee, 2010). The research framework was also adapted from Hevner, from his A Three Cycle View of Design Science Research, which analyzes design research science with a help of three cycles of activities (Hevner, 2007). Other literature, such as books and articles were used to support the theory presented in these books.

The second most used subject for literature discussed version control systems, and their use as a course platform. A study compiled by Haaranen & Lehtinen (Haaranen & Lehtinen, 2015) regarding the usage of GIT as a course platform led the researchers of this thesis to familiarize themselves with other studies where the usage of version control system as a study platform was studied (Biňas, 2013; Kelleher, 2014; Kertész, 2015). Fair amount of studies regarding VCS usage as a study platform have been released over the recent years. Using VCS as a study platform could be considered as one of the development objects, therefore the same subject will be a part of this study too.

The third literature subject discoursed on process management. The second study addressed the problems with the project preparations and processes completed during the preparations. During the second study, the research focused on practices, how the processes could be optimized so that the costs during the project could be reduced. The success of a project should not be measured merely by financial facts (Davenport & Beers, 1995, p. 58) but rather with other factors, such as customer satisfaction. When the processes are optimized, the project proceeds fluently and the product, this case a new VCS can be made available to the customers without unnecessary delays.

1.6 Evolution of VCS

The trajectory of version control systems can roughly be divided into three different generations, 1st generation, 2nd generation and 3rd generation. First version control systems were developed in the early 1970's when Source code control system (SCSS) was released. It was developed to help programmers control changes in the source code currently in development (Rochkind, 1975, p.364). Since then there have been a great number of version control systems and the current systems in daily use represent the 3rd generation of the software.

The era of 1st generation systems spanned from early 1970's to mid-1980's. Typical for these systems were that the software stored data locally on computer and used locking method as a conflict resolution.

The 2nd generation systems started to arise at the mid 1980's, replacing the old 1st generation software. The era of the 2nd generation VCS continued till 1999. First feature that the 2nd generation systems introduced, were a centralized client-server data model which required an active network connection to function properly. The usage of the 1st generation software required that all developers of a project had to be on the same machine as the single central project repository, and the 2nd generation software changed that by allowing developers to access the single repository from another machine over the network. The second introduced feature was merge before commit as a conflict resolution. In the 1st generation software, the files were in read-only format and when someone wanted to edit a file the system would make the file writable and lock it, so that no one else could edit it at the same time. In the 2nd generation software, the system noticed when a file has been changed during the time a person has been editing it and requires that the conflict is resolved before the file can be saved. The reason for this is that someone else edited the file at the same time and saved the file (Raymond, 2007; Ruparelia, 2010, p. 5).

The 3rd and the current generation of VCS began in 2000, when Bitkeeper was released. Since then, several different version control systems have been released which all have some different functionalities and target audiences. The biggest overhaul when comparing 2nd and 3rd generation systems is the decentralized data model. In a decentralized repository model, both, the server and the client computer has a version of the repository. As the client computer, has also an offline version of the repository, files and documents stored on the local repository can be edited even when the computer does not have access to the network. The changes can be committed to the repository on a server when the computer is connected to the network. The conflict resolution method as also improved between generations. When the 1st generation systems locked the file when a user was editing it, the newer systems allow multiple users to access and edit the same document the same time. In multiple edit – scenarios, when user tries

to commit changes to the repository at the server, the system will inform user that the repository on the server has already been edited and the system will allow user to merge the changes he has done to the repository on the server (Sink, 2011, p. 1).

A list of some of the most notable version control systems is shown in the Table 1.

System name	Release year	Conflict resolution	Model
SCCS	1972	Locking	Local access
RCS	1982	Locking	Local access
CVS	1985	Merge-before-commit	Centralized client – server
Subversion	2000	Merge or locking	Client – server
Bitkeeper	2000	Commit-before-merge	Distributed
Mercurial	2005	Merge	Distributed
GIT	2005	Merge	Distributed

Table 1: Some of the most notable VCS releases (Chacon & Straub, 2014; Raymond, 2007; Tutorial-Conflict, 2013)

Table 1 presents some of the most notable version control systems and the year were released. The table also shows the evolution of conflict resolution method, the early systems prevented conflicts by locking the edited file for one editor, current systems have built-in merge functions which can be used in a case of conflict resolution.

1.7 Research questions, unit of analysis and objectives of the study

A research question is a statement that announces the phenomenon that will be studied to understand it. This was the very first that was decided at the beginning of the study. "The first step in the survey process is to determine the research objectives. The researchers should agree on a well-defined set of research objectives. These are then translated into a set of key research questions. For each research question one or more survey questions are then formulated, depending on the goal of the study." (De Leeuw & Dillman, 2008, p. 4.)

In this study, the research question is "How to comprehend, rationalize and develop version control systems?" the answer to this question was found with a group of smaller questions asked during the data collection process. "Research questions are usually distinguished from the questions researchers actually ask participants in interviews or during field observations or ask of data during analyzing them. Although the questions participants answer and the constant questioning process that defines qualitative data collection and analysis are in the service of answering research questions, they are not equivalent to them." (Given, 2008, p. 787.)

When analyzing the research data, the first step is choosing the unit of analysis. This will tell how the data should be analyzed. It is the phenomenon that is analyzed in the study. The unit of analysis was quite clear after the main research question was set. As Yin stated in his book "Selection of the appropriate unit of analysis will start to occur when you accurately specify your primary research questions." (Yin, 2009, p. 30.)

The unit of analysis in the study is a version control system. In a qualitative research process, all the smaller studies can have their own units of analysis, which will help to form the primary unit of analysis. "As a general guide, your tentative definition of the unit of analysis (which is the same as the definition of the "case") is related to the way you have defined your initial research questions." (Yin, 2009, p. 4.) In this case the unit of analysis is the same during the whole study and it was used in the analysis phase of the data collection processes of the smaller studies.

The objectives of this research were to discover answers to the research questions represented in each of three special questions. The research questions are listed below, organized by special questions. Answers to the questions 1 and 2 were examined by both researchers and both researchers had their own question 3, which they examined separately.

Special question 1:

1. How could a survey be used to improve the usage of version control systems in Aalto University?

Special question 2:

2. What are the preparations prior a version control system project?

Special question 3:

3. How can the test results be used to improve the user experience?
4. How should the testing be performed during a VCS project?

2 Research methodology

In this chapter, the researchers will discuss the methodologies utilized during each step of the project. Methodologies utilized in each process, will also be assimilated to the research literature published about each methodology to justify the usage of chosen methods during the study in hand. The four most fundamental methodologies applied in the study will be presented later in Chapter 2.1.1.

2.1 Purpose of the study

The purpose of this study is to understand the phenomenon called a version control system (later abbreviated as VCS). The study aims to understand the VCS in theory and how could a new VCS be acquired and integrated into a current IT environment and what kind of procedures

that inquires? Rationalization for the study was to determine the best practices for implementing a new system alongside existing systems.

The project consisted of several different tasks which all relied on different types of methodologies. In the following chapters, we will explain the sub-projects and methodologies used in different projects.

In the future, the results of this study can be utilized during similar projects to avoid possible problems appointed in this study. With the help of the results of this study, project managers can get acquainted with the methods found useful in both data gathering and processing.

At the beginning of the study, the data collection was decided to implement with a qualitative research method. Since the focus of the data was in quality instead of quantity. John Creswell defines qualitative research in the following way: "Qualitative research is a means for exploring and understanding the meaning individuals or groups ascribe to a social or human problem. The process of research involves emerging questions and procedures, data typically collected in the participant's setting, data analysis inductively building from particulars to general themes, and the researcher making interpretations of the meaning of the data." (Creswell, 2013, p. 4.)

The study was composed by combining several smaller studies. During the study, the staff and administrators of Aalto University were interviewed and based on the interview results in a VCS project was engineered. The study will also debate on the processes adopted before and during the actual VCS project.

2.1.1 Research approach

The theme for this study is to comprehend, rationalize and develop the usage of version control systems at Aalto University.

The methodology for the entire study is case study research. The case study consisted of several minor studies which all applied their own methodologies. The common nominator for all the separate studies is a qualitative approach to the study results. As the studies were conducted as a qualitative study, the research approach to the studies was a general inductive approach. Per David R. Thomas, general inductive approach can be seen as a simple, nontechnical method for analyzing qualitative data (Thomas, 2006, p. 245). The aim of using a general inductive approach in this study was to make broad generalizations based on the study results discovered during each individual study. Based on the generalizations, the main theme of the study could be answered.

The entire study was a compilation of three minor studies. The results of the minor studies were analyzed to answer the research question of the main theme of the study. The most fundamental methodologies and use of purpose of using them, alongside the ones mentioned earlier, will be described in the following paragraphs.

The first study was conducted to understand the current situation of version control usage at the university. This was a focus group research, which was conducted by making semi-structured interviews. The results were used to supplement a preliminary report conducted before the project. Focus group research suited well for the purposes of the semi-structured interviews as focus group research encompass a wide range of discursive practices, spanning from structured interviews with delimited topics to open-ended conversations (Kamberelis & Dimitriadis, 2013, p. 4).

The second study focused on the preparations of the version control system acquisition project. The second study was a participatory research where the data was gathered with literature and by taking part to the tasks that were done during the preparations. Participatory research emphasizes the participation, contribution empowerment and emancipation of all relevant parties in examining a common problem (Reilly, 2010, p. 558). The tasks that were examined during the preparation were market analysis and feasibility study where the possible future systems were inserted into a matrix and compared against the requirements which were based on user interviews and the preliminary report.

The third study discoursed the testing methods and the benefits of testing in the VCS project. The answer to the research question was sought by comparing the testing methods used during the project with studies released about testing methods. The testing phase in the project could be compared with several of the usability inspection methods presented by Jakob Nielsen in his study released in 1994 (Nielsen, 1994, p. 413). The pluralistic walkthrough method could be compared to the testing method in the VCS project as the testing was conducted with the help of user stories written before the actual testing, the difference being that instead of reading the cases out loud, each tester conducted the tests by themselves.

The testing used the iterative waterfall model as a framework. The framework was a natural choice as a framework because testing was conducted by the same person, who will be responsible for the system in future. While testing was an iterative phase, errors discovered during the test could be revised already during the testing phase.

2.1.2 Framework applied in the research

The framework applied to the study as an analytical tool was Alan R. Hevner's Design Science Research Cycles (Hevner, 2007, p. 2), seen in the Figure 2.

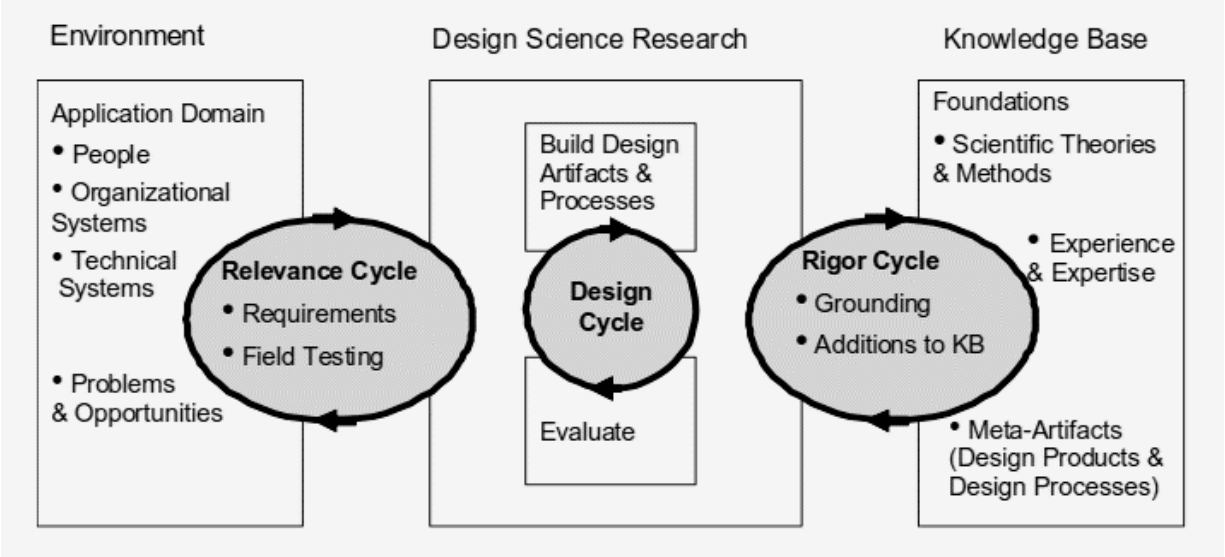

Figure 2: Design Science Research Cycles per Hevner (Hevner, 2007, p. 2)

The framework consists of three cycles, which are used throughout a design research project. The arrows in each cycle symbols an iteration process. Iteration is used until the outcome of each process fulfills the set demands.

The relevance cycle connects the specific environment with the research. It is used to examine the environment's requirements and for the field testing of the product, in this case the VCS. The design cycle iterates between product building and its evaluation and the rigor cycle connects the knowledge database of scientific theories, experience and expertise to the research and product building. The cycle model is used from the beginning of the study and it is used until the new product is ready to use.

2.1.3 Triangulation

Lisa M. Given states in the book she has edited the following "Triangulation in qualitative research has come to mean a multimethod approach to data collection and data analysis. The basic idea underpinning the concept of triangulation is that the phenomena under study can be understood best when approached with a variety or a combination of research methods." (Given, 2008, p. 892.)

The types of triangulation that were used were multiple investigator triangulation, theory triangulation, the triangulation of methods and triangulation of data sources (Given, 2008, p. 893).

- In multiple investigator triangulation, more than one investigator collected and analyzed the data.
- In theory triangulation, the researchers examined data from multiple theory sources.
- The triangulation of methods, the researchers used multiple research methods.
- During the triangulation of data, the researchers collected data from several sources, such as written documents and interviews.

The theory triangulation examined the theory of version control systems with multiple literature resources, such as Pro GIT (Chacon & Straub, 2014) and Version Control by Example (Sink, 2011). With the help of the theory sources, the researchers acquired an understanding of the version control systems, how they function and how they were evolved during the years. Multiple sources were used to avoid a unilateral and concise view of the systems.

In triangulation of methods, multiple research methods were used to acquire an adequate amount of explicit data. These methods were also used to avoid making incorrect assumptions and other errors. This way the research process proceeded logically and with right amount of pace. These methods included such research methods as case study, a market survey, semi-structured interviews, surveys, qualitative data analysis, theory analysis, feasibility study and participation in to the VCS project.

In the triangulation of data, Multiple source of data were used to provide acceptable amount of research data and extensive conception of the studied phenomenon. This provided more extensive and more exact results and this effected positively to the credibility of the results of the study. Per Runeson & al. different types of evidence, figures, statements, and documents are combined to support a strong and relevant conclusion (Runeson & Höst, 2009, p. 16).

2.1.4 The background and roles of researchers

The study is partly compiled by two students of Laurea University of Applied Sciences, who both also work at the Department of Information Technology at Aalto University. The study consists of case study research questions. Two out of three questions were made jointly and the third question was separately done by the writer of this thesis. While the thesis was written, both worked as a part of the version control system acquisition project which is also the subject in this thesis. The introduction, methodology and description of operational environment were compiled together, the rest of the thesis was done separately.

As both were part of the VCS project, it was a natural decision of studying the project together and compile the thesis based on the observations made during the project. Since the project was made during normal working hours, neither of them participated in all the processes of the

project, because they also had their normal work assignments. The parts that they couldn't participate in are also mentioned in this study.

The researchers of this study participated in most of the steps of the project. The only step that they didn't participate was the actual VCS installation process. Researchers were also part of the preliminary project group which composed the preliminary data.

Though both researchers had solid knowledge in the field of information technology, the project could be considered as their first substantial project they participated actively from the beginning to the end. Besides working for the project, the researchers had to familiarize themselves in the effective project working during the lifecycle of the VCS project. "To be reliable and fulfill project objectives, the researchers must be trained how to document his or her observations and must have a strong understanding of research goals."(O'Grady & O'Grady, 2009, p. 34.)

Anne Sofia Fink divides the research process into seven stages in her article The Role of the Researcher in the Qualitative Research Process. A Potential Barrier to Archiving Qualitative Data. As reported by Fink, those stages are thematizing, designing, interviewing, transcribing, analyzing, verifying and reporting (Fink, 2000, p. 198). The researchers had a role in each of these stages. The stages of the research process are described in Table 2.

Stage	**Task for the stage**
Thematizing stage	Decision about what will be studied, why and how it will be done
Designing stage	The planning of methodologies that will be used in the study. How the interviews are going to be done, how will the data from the interviews be transcribed and analyzed, how the data will be verified and reported.
Interviewing stage	Research data collecting by interviewing the people who use VCS systems
Transcribing stage	Interviews will be transcribed
Analyzing stage	Data collected during the interviews will be analyzed
Verifying stage	The results will be verified against the field notes and interview recordings
Reporting stage	Results of the interviews will be reported

Table 2: Stages of the research process

2.2 Data collection

As Kumar Singh states, behavioral science data gathering is generally done in various ways. (Singh, 2006, p. 218). As this study is a result of several smaller studies, the means of gathering data varied quite a bit from study to study. Eisenhardt debates also on the data gathering during case studies, stating that collecting data during case studies often relies on interviews, questionnaires and observations (Eisenhardt, 1989, p. 534).

The preparations for the preliminary project started in March 2015 and the kick-off day for the actual project was in September 2015. The following studies were conducted during that time.

1. A preliminary report (n=1) was composed.
2. The preliminary report was complemented with the results received from the case study (n=1) based on user interviews.
3. Based on the enhanced preliminary report a comprehensive market analysis was done by the administrators of Helsinki Institute for Information Technology (later in the text abbreviated as HIIT).
4. Study about preparations prior to the version control project was conducted and reported.

A test phase was also included in the project. During the test phase, the project group mainly focused on non-functional tests. A more descriptive description of the steps taken before the official project kick-off is presented in Study II: Study about preparations prior the VCS project. All the three conducted studies required and produced vast amount of data. The data used and gathered are listed and described below:

Document	**Content**
The preliminary report (n=1)	The preliminary report was conducted by the manager of the project before the actual project started. The preliminary report included statistics regarding the current usage of version control systems in Aalto University. Report also included preliminary requirements for the future system, information about the service concept and resource requirement estimation. The preliminary report was later supplemented with the customer survey findings.

Meeting transcripts (n=15)	Project meetings took place weekly at the beginning of the project, after that, meetings were arranged when necessary. All the meetings were documented and the documentation was used in study 2 when the preparation process was evaluated. The evaluation focused on the meeting transcripts composed during the preparation stage.
Interviews of administrators and users (n=3)	Interviews were conducted at the beginning of the preparations of the project. Two researchers conducted the interviews. Each of these interviews produced a data document, which were then compiled into a single document.
Product comparison (n=2)	Product comparison consisted of two different documents, the first document was composed by the administrators of HIIT, the document described the current version control system usage at HIIT, requirements for the future systems and suggestions for the future system. The secondary document was composed by two researchers. The second document focused on the product comparison of selected VCS's and examined their features and functions and compiled a document based on the results.
Project plan (n=1)	Project plan was made by the project manager during the preparations of the project. It had a rudimentary plan for the tasks and progression of the project.
Suggestion of investment for the university (n=1)	Suggestion of investment was made by the project manager during the preparations of the project. It suggested that the university should invest in the acquisition of GitLab's community edition. The suggestion had 11 different categories, that explained, for example, the concept and functionality in the environment and benefits of investment.

The risk analysis of the project (n=1)	The risk analysis of the project was conducted by the project group during the preparations of the project. It produced a risk assessment table, which had the name and subscriptions of each possible risk.

Table 3: Documents gathered and used in the studies

2.2.1 Preliminary report

The data for the preliminary report were gathered with methods similar used in focus group research. Focus group research was a suitable method for the study "Focus groups are ideal for exploring people's experiences, opinions, wishes and concerns." (Kitzinger & Barbour, 1999, p. 5.) Data gathering methods resembled ways which data are gathered in focus group researches, but in Aalto's case, instead of having 3 to 5 bigger groups where the user would freely discuss the subject, we had 3 smaller groups which contained the administrators of current systems (n=9). Administrators discussed freely about the current systems and possible future centrally managed system. A preliminary report was conducted based on these group meetings. The preliminary report also contained requirements received via email from the users of the current VCS's.

2.2.2 Case study

According to Benbasat & al. there are three reason why case study research is a viable information system research strategy. Firstly, the researchers can study information systems in their natural environment and create theories from practice. Secondly, the case study allows researchers to answer "why" and "how" questions and thirdly, a case study is an appropriate method of conduct to research in an area, which has been studied rarely in the past (Benbasat, Goldstein, & Mead, 1987, p. 370).

The case study process adapted Yin's case study research process (Yin, 2009, p. 1). Steps of the process Yin introduced in his study, can be seen in Figure 3. Yin's process starts from planning the study, and proceeding from there onwards until the study results are shared. Yin's process is somewhat agile; some phases of the process can be overlooked, but it is always possible to go back to the disregarded step to carry out that step with newly gained knowledge.

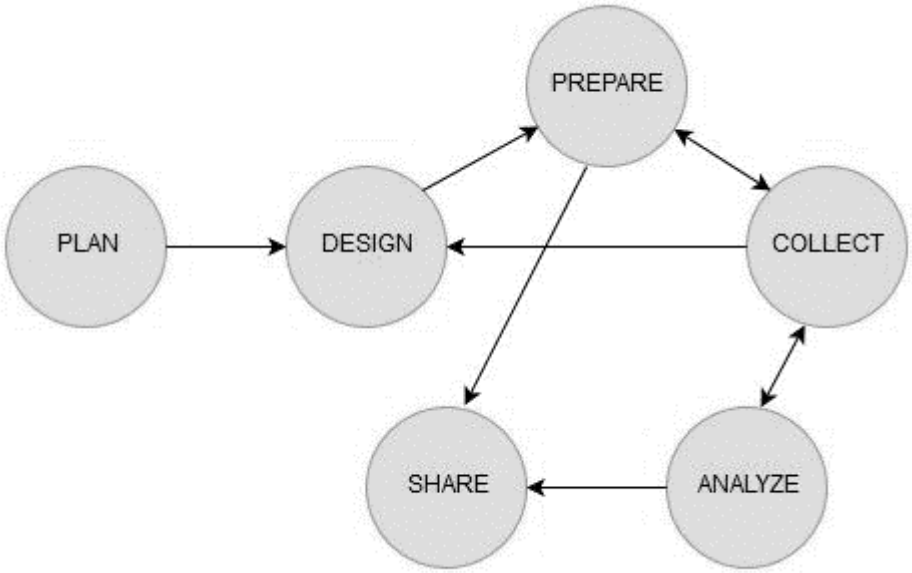

Figure 3: The case study Research Process (Yin, 2009, p. 1)

Study was performed by interviewing the users and administrators of current systems (n=4). The data gathering in the case study was done in a very similar way described by Kaplan & Duchon in their own system acquisition - study (Kaplan & Duchon, 1988, p. 577). The results of the interviews were analyzed and then they were presented to the project manager from the university's centralized IT department's project office who oversaw the version control development program at the university.

The interviews were done as semi structured interviews. "Like the structured interview, the semi structured interview tells you how respondents answered the standard questions. In addition, the semi structured interview allows the investigator to ask additional questions to follow up on any interesting or unexpected answers to the standard questions."(Mitchell & Jolley, 2012, p. 277.) The interview process was divided into four stages:

1. Requests for interviews were sent
2. The interview questions were send in advance to the interviewees
3. Interviews were held
4. Transcription of the interviews.

During the interviews, the interviewer wrote down the answers and recorded each conversation. The researchers transcribed the interviews afterwards. The recorded audio-files were used for this and were deleted after the process.

As a part of the case study, two researchers also familiarized themselves with the available documentation about version control systems. The unit of analysis in this case study was a version control system.

2.2.3 Market analysis

Market analysis was performed by one of the administrators of HIIT (n=1). As the department had a similar ongoing project, centralized IT could take advantage of the market analysis they had already compiled.

The methodology used in market analysis was evaluative research. "Evaluation research is applied in that the aim is to produce knowledge that will contribute to greater understanding of the effect of a defined activity. This activity may be referred to in many ways, such as intervention, initiative, and policy." (Given, 2008, p. 303.)

Certain requirement points were discovered before the analysis, and by comparing available products and the features provided by the products, HIIT administrators could determine the program which would be the most suitable for their use. The evaluation process was done in a very similar way as Ronald. R. Powell explains in his study which was composed for libraries (Powell, 2006, p. 115). As the project was like the one which was already in progress by centralized IT, HIIT abandoned their own VCS project and started to collaborate with the centralized IT.

2.2.4 Project preparations study

The study about the preparations before the actual project was conducted by two project group members from the IT customer services of the university (n=2). The research question for this study was "what are the preparations prior to version control project". The data for the study were gathered by using two different methods. The two members participated all the meetings that the project group had before the project kickoff and took notes. They also carried out several preparation tasks, such as feasibility study, themselves and made documentations about them.

During the feasibility study, several different version control systems were chosen for comparison. Comparison was done by placing the chosen systems in a matrix where the requirements for the new system were also placed. By comparing the systems against the requirements, project team could narrow down the options and decide the most suitable version control system for this project.

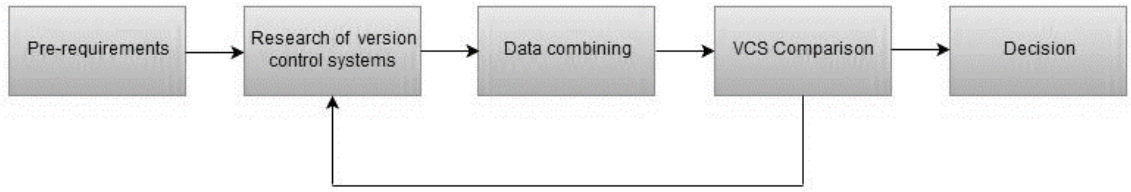

Figure 4: Comparison process

Figure 4 demonstrates the process during the comparison of version control systems. At first, the pre-requirements for the system were clarified. After the requirements, had been organized into a matrix, a set of possible future systems were organized into the matrix alongside with the pre-requirements. At this stage, the requirements and possible systems were in the same matrix, this made it possible to rank systems against the requirements. The system, which received most points, was chosen for a feasibility study.

A framework for feasibility study was also used for this study. The framework has two different divisions, concept planning and feasibility analysis. The concept planning was dispensed and study begin from the beginning of feasibility analysis. In this research, product comparison was a part of the feasibility study, after comparison process, the product chosen underwent the evaluation steps similar to the feasibility framework shown in Figure 5.

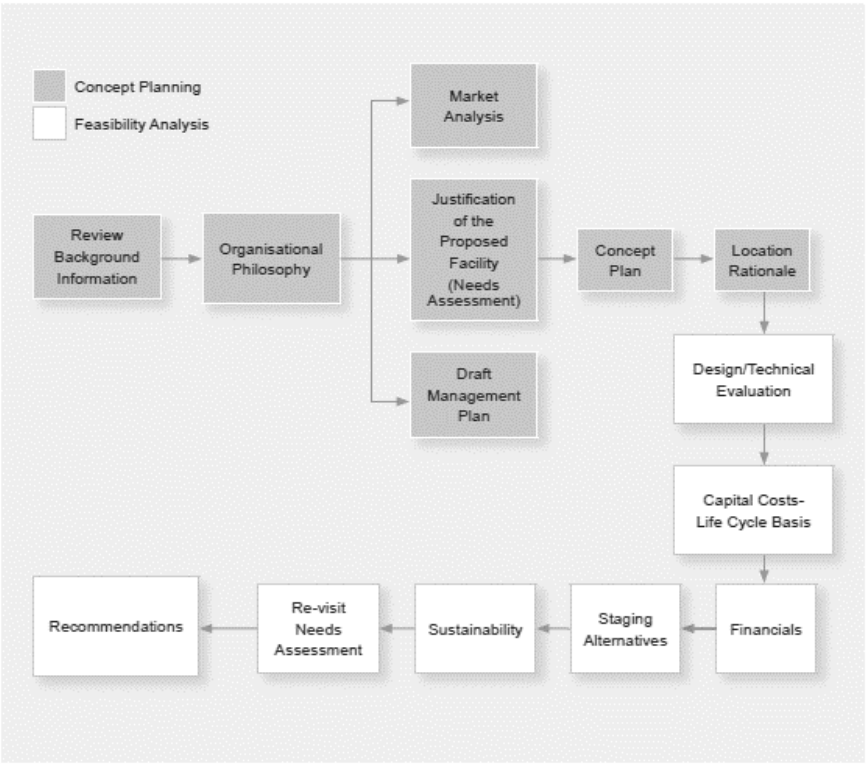

Figure 5: Framework for feasibility study (Didcoe R., July, 2007, p. 7)

During the feasibility study, the project group received a design & technical evaluation from the information security group of Aalto IT. The project manager evaluated the financial and sustainability possibilities and after that he made an acquisition proposal for the system to Aalto University's SGPP. Project preparations study was made solely to find an answer to the main research question.

2.2.5 IT security requirements

The IT security requirements of the VCS project was examined 4 times during the project. The requirements need to be fulfilled, so that a new system can be integrated into Aalto's IT environment.

The questions asked during the process were, what are the IT-security analyzing methods for a new version control system and what is the analyzing process when comparing the security features of a version control system with the university's IT-security requirements. These questions were asked to find an answer to the main question.

First the project group needed an IT security statement for the statement of suggestion. For this statement, the project group formed a small subgroup of 3 people to conduct a small study about GitLab's security features (n=3). Both GitLab Enterprise Edition and Community Edition. The security features were gathered from the GitLab Documentation found at GitLab's homepage. After the study, the results were presented to a person from IT security group (n=1), who then gave the statement.

Second step was the risk analysis of the project. This was put into practice soon after the project kickoff by the whole project group (n=5). It was done as a workshop, where the possible threads were identified, discussed, analyzed and the possible solution were figured out. The project group used a risk assessment table as an evaluation tool. The group could name the risks, write a detailed description of each of them, calculate the risk priority, write down recommended actions for each risk, name an owner to a particular risk, update the risk status and when the information about that risk was last updated.

As a third step, the subgroup analyzed the information security of GitLab (n=3). This was done by comparing the features of the system with the IT security framework of Aalto IT. The group used basic level demands of the framework and it was edited so that all the unnecessary demands were deleted. After this GitLab's features were comparted to the framework's demands. GitLab passed all the necessary requirements.

Finally, after the project group had done all its own security tests with the VCS in test environment, the IT security group conducted its own tests. They tested most of the functions and

features of the forthcoming systems and tested the security features with a series of random testing. GitLab passed all the tests.

2.2.6 Test phase

Testing during the project was mainly focused on the nonfunctional requirements, but before nonfunctional requirements could be tested, a set of functional tests had to be run to determine, if the software is working as desired.

As the version control system, which was chosen during the feasibility research, was already well-established, the project group decided to focus the testing on nonfunctional functionalities, which per Dean Leffingwell, can be divided into qualities such as usability, reliability, performance, supportability and so on (Leffingwell, 2010, p. 75).

Usability and reliability were tested with a group of user stories developed for the testing purpose. Testing was completed in two different batches, the first batch was completed in the test environment, and after the issues found during the testing had been redeemed, the system was transferred to another server which will act as production environment and the second batch of testing was conducted.

The methodology used during the tests, and the whole of the project, could roughly be compared with the waterfall method of software development. Once a step in the model was completed, results were evaluated and if the evaluation was passed, the project was ready to move to the next step. If a certain step failed, project group started over the current step from the situation it had been developed into, and correct the malfunctions detected during the evaluation. When looking at the steps, the VCS project, and especially the testing phase was very much an iterative waterfall.

These tests were a crucial part of the installation process of the new system. Not only would they give valuable information about possible system flaws, but can also help to develop the system in the future. Jay Nunamaker agrees with this and brings this out in the study, while also adding that in addition to system development, testing and observing a new system can even lead to a discovery of a new theory of the system (Nunamaker Jr, Chen, & Purdin, 1990, p. 100). In Aalto University's VCS project, issues found by the testers were redeemed and in addition, based on the test results, few additional functionalities were added to the system.

2.2.7 Data analysis

As a result of the data collection studies, there was a great amount of raw data.
The raw data of each study were analyzed and categorized by using the chosen unit of analysis. The analysis had 4 phases.

1. The raw data were taken apart to smaller units of data
2. The smaller units of data were analyzed
3. The units with the same or similar data were categorized
4. The data were consolidated

The analysis of the results was a necessary part of the study to understand and answer the research question. "The process of data analysis involves making sense out of text and image data. It involves preparing the data for analysis, moving deeper and deeper into understanding the data, representing the data, and making an interpretation of the larger meaning of the data." (Creswell, 2013, p. 217.)

Data analysis can be seen as process, where the data gathered are interpreted in the way they will make sense (Merriam & Tisdell, 2015, p. 175). The data acquired from the survey process had to be processed into a form, where it would be easy to interpret and at the same time the integrity of data would not be compromised. To retain the integrity of data, each interview (n=3) conducted during the case study was transcribed from the recordings of interviews into a written document.

By transcribing the interviews, researchers could debate on each interview question one at the time, making it also possible to compare the answers of each question between separate interviews on a literal level. The importance of transcribing is pointed out by Sharon Ravitch as she describes transcribing by saying that transcribing is not only important for data collecting, but transcripts are also useful for valid and rigorous data analysis (Ravitch & Carl, 2015, p. 157). After the interviews were transcribed, the data were compared with each other. The challenge of comparing data was to get all the data into the same form, in this study this challenge was conquered by conducting interviews in a semi-structured model (Folkestad, 2008, p.10). The results of the common attributes identified by this method can be seen in paragraph 2.2.8.

The data that were analyzed after the second study consisted of meeting transcripts, field notes composed by two researchers, literature and studies published about each task conducted during the preparation period. Furthermore, the observations done during the preparation period were also included in the data. As the data were collected from multiple sources the researchers ended up evaluating the data with evaluative methods.

The researchers analyzed the preparation tasks from the viewpoint of time-efficiency and necessity. By doing this, the researchers aimed to optimize the tasks which could be a part of similar projects in future.

After the preparation tasks, the field notes composed by the researchers were organized according to tasks. The field notes were supplemented with meeting transcripts. After combining the data, each researcher familiarized themselves with the literature and studies available from each subject. By comparing methods used in studies and subjective information regarding each task, researchers could complete their perspective for optimized tasks.

The study which addressed on testing at the VCS project relied on the testing data and the observations done during the actual testing. The data gathered during the tests were entered into the same Excel document where the original test cases were designed. The data were then analyzed and the problems discovered during the analysis were reported to the technical administrators of the newly acquired system. The test results were also rationalized to answer both research questions:

1. How should the testing be performed during a VCS project?
2. How can the test results be used to improve the user experience?

By rationalizing the data, in this case results and observations, researchers could confirm the hypothesis that testing, and test results are a vital part of system development.

2.2.8 Attributes identified in study

Though the person, who were interviewed, had a different point of view about the usage of VCS, researchers could discover at least two common attributes that were mentioned by everyone during the interviews. All the phases from the interviews to the identified attributes, are described in the Table 4.

Phase	**Process**	**Result**
Phase 1	Reading of the interview notes, hearing of the interview recordings	Understanding of the interview answers
Phase 2	Fulfilling of the notes with the information from the recordings	Answers have more details
Phase 3	Dividing of the answers according to their themes	Several different themes, waiting for a deeper analysis

Phase 4	Combining and editing of each themes' answer material	Combined and edited answers of every theme
Phase 5	Analysis of the combined answers of each theme	Rationalization of the answers
Phase 6	Possible rollback to phase 1, if the analyzed answers are insufficient	Better understanding of the interview results
Phase 7	Composition of attributes	Common attributes that were discovered from the interviews

Table 4: Phases from the interviews to attributes

The identifying process started by the iteration of the interviews, filling the notes with details that might have been missed during the interview sessions. Then the answers were divided according to their themes and after that combined, edited and analyzed. In case that the analysis would have given insufficient data, the process would have rolled back to phase 1 and continued from there. Finally, the attributes were composed. The recognized attributes are displayed in Table 5.

Subject	**Common attribute identified**
Repository management	Normal user should be able to create repositories. User should also be able to delete repositories which are no longer used.
Accessibility	The version control system should be accessible outside university's network. System should be accessible with the most common operating systems & devices
User management	Owner of a repository should be able to manage user rights to the repository he owns. Owner should be able to set the repository as public or private, this should also be possible throughout the whole lifecycle of a repository.

Table 5: Common attributes identified in interviews

3 Phases of the study

In the following chapters, the researcher will discuss the data collection process for this case study. As a research method, case study has received frequent criticism for the fact that case studies often rest on the results of a single case (Dubé & Paré, 2003, p.609). To avoid the single case criticism, the data for this study were collected with three separate studies. This chapter describes what these studies were about and how they would help to answer the research question of this study. All three studies were part of the version control system acquisition project.

3.1 Study I: Customer demand survey

The aim of the first study was to find answers to following research question: How could the usage of version control system be improved at Aalto University. Answer to the research question in this study was sought by conducting user interviews and interpreting the interview results against the literature released about the subject.

3.1.1 Goal of the project

The primary goal of the project was to acquire a centrally managed version control system and provide it for the departments which were hosting their own version control systems when the project started.

When the study was launched, there were approximately 900 active version control system users at Aalto University. Many departments hosted their own systems, some of them were even hosting multiple systems at the same time, depending on the preferences of the researchers. Additionally, the new system would be made available to all Aalto users and cooperation partners participating in projects with Aalto researchers.

The new system is hosted by Aalto University's IT department. By hosting the new system, IT department will release resources from other departments as the new system was acquired to substitute overlapping systems hosted in the departments.

3.1.2 Benefits of the project

The most important benefit of the project was that the number of version control systems in the university decreased, the number of the people using version control systems increased, the number and quality of version control system documentation increased and the university got more tools for coding environment. A centrally managed VCS also released resources as the local administrators no longer had to update and maintain the servers where the current systems were hosted.

As mentioned before, one of the main targets for the project was to provide VCS, which users could use with centrally managed credentials. This made cross-department co-operations easier as all the students and staff members receive centrally managed credentials upon entering Aalto University.

3.1.3 Current usage of the system

The project manager composed a preliminary report of the systems which were used at Aalto University prior to the project start. As the preliminary report was composed purely based on opinions and observations, the report was supplemented with a case study where the users and administrators (n=4) were interviewed by the researchers. The interviews were done as open interviews; users were sent a few questions before the interviews but the interview itself was not guided by these questions but the conversation was quite free and the questions sent in advance were only used if the conversation wouldn't advance without external questions.

3.1.4 User interviews and observations

As a part of the case study, researchers contacted local administrators and asked from them, if their department used version control system and if yes, could they point us to an active user of the system who we could interview regarding the software usage habits. The interviewed administrators represented Helsinki Institute for Information Technology (HIIT), Department of Mathematics and Systems Analysis and Department of computer science. The following questions were asked from users:

1. Which revision software tools are you using?
2. Are you using more than one revision control system?
3. How often do you use revision control tools?
4. For what purpose, do you use revision control?
5. Which functions of the system are necessary for you and which are essential?
6. How could (or should) the system be improved?
7. Is the system you are using, easy to use? Have you encountered problems in system usage, if yes, what kind of problems?
8. Would you be willing to change the revision control system you are using to another system?

The questions were sent in advance to the interviewees and they were asked to read the questions before the actual interview. This gave the interviewees understanding about the nature of the up-coming interview. They were also given an option to literally answer the questions and return the answers to interviewers. The main reason for this was the timetable of the

interviewees. Fortunately, all who were contacted wanted to participate the research and made the time to their schedules for the interviews. Nobody wanted to send literal answers.

The interviewees were also asked if they wanted to give the interview in their own environment or at some other place at the university. Everyone wanted to give the interviews in their own offices. This affected the nature of the interviews. The interviewees talked freely about the subject and the interviews were held as an open interview.

During the interview when there was one interviewer and two interviewees the interviewer wrote down notes at the same time he was speaking with the interviewees. During the rest of the interviews one of the interviewers made notes with his computer and the other interviewer talked with the interviewee. All the conversations were also recorded with a dictation machine. The audio files were listened to later and compared with the notes that were made during the interviews. All the missed details were written down afterwards. Prior to the interviews, everyone was asked for a permission to record the sessions and everyone granted the permission for recording. They were promised that only the interviewers will use the audio files in this case study and that the audio files will be deleted when they will no longer be needed.

The common attributes identified during the interviews can be found in Chapter 2.2.8, therefore in this chapter, we will focus more on the technical observations and the use of version control systems on department premises.

Version control system is widely used among the departments of Aalto University, at the Department of Computer Science, VCS has even been ported as a course platform (Haaranen & Lehtinen, 2015), which allows teachers and students to collaborate while the VCS can be used to distribute course materials and tasks. VCS as a collaboration tool, has received fair amount of recognition in the field of information technology studies. Many companies except that their workers are able to use collaborative tools and VCS is among these tools (Kelleher, 2014, p. 1; Kertész, 2015, p.381), therefore it is a vital skill to learn before entering the working life.

Other observation made during the interviews is that different systems have different uses, while the older system, SVN, is more popular among people who use VCS while writing conference proceedings, the newer system, GIT, is popular among researchers whose VCS usage is based solely on storing and editing software code on the system. According to the interviews, SVN and GIT users can also be divided into groups based on the age of the user, older researchers are found of SVN, younger researchers prefer GIT.

3.1.5 Answering the research question

As mentioned before, the research question for this study was "How the usage of a version control system could be improved at Aalto University". By analyzing the customer survey results, the question can be answered and the future system can also be improved based on the interview results. The usage will be improved when the amount of overlapping systems is reduced.

When the new system is administrated by centralized IT and used with credentials which are provided for all Aalto users, users can collaborate with other departments more effectively as they no longer need to acquire separate credentials for the collaboration platform. When certain functionalities will be introduced to the centrally managed system, departments can utilize the new system also as a course platform. Once the students and staff are familiar with the new system, it can be used effectively to distribute course material to the students and students can use the platform to return assignments to the teachers.

When there is only one system in use, support for the usage of that system can be enhanced as users and people who offer support for the system, can concentrate on a single system instead of learning several different systems. When the user support can be transferred from administrators in the department to the support people in centralized IT, the administrators can focus more on the development of the system instead of providing user support.

3.2 Study II: Study about preparations prior the VCS project

The customer demand survey was only small part of the preparations done prior the version control system project. The second study was conducted after the project had already started, the aim of the study was to review the steps done during the project preparations. The research question for the second study was "What are the preparations prior to a revision control project?"

To answer the research question, researchers had to analyze each component finished before the actual project started. The documentation of this study will progress in chronological order.

Preparations for the project begun in mid-March 2015 and the project kickoff was at the end of August 2015. During that period the project group had 13 meetings. "During this phase, qualitative researchers plan all tasks, responsibilities, and time estimates associated with the project" (Given, 2008, p. 688).

The most important tasks finished during the preparation time were:

1. Forming of the project group.
2. User interviews.
3. Product comparison.
4. Feasibility study.
5. Project Plan.
6. Suggestion of investment for the university.
7. Risk analysis of the project.

3.2.1 Project group

The project was executed by the centralized IT departments' project office. The project manager was chosen from the project office and he chose the rest of the project group. The members of project group were chosen from the different groups of the university's centralized IT department. Also, a member of Helsinki Institute for Information Technology, HIIT joined the group, because HIIT was looking for a replacement version control system to their current system.

The criterion for the project members was necessary knowledge from project work, IT architecture of the university, initialization of services, information security, revision control systems and authentication methods.

The group was built from people who work for the centralized IT and HIIT. The project manager was provided by centralized IT, one of the members worked as a Linux specialist, two participants worked in the customer service and the last participant was an administrator for HIIT.

The project manager has worked in centralized IT's project office for several years and is experienced in similar projects. He is responsible for planning and leading the project, communicating with various groups inside the IT and communicating with the external interest groups. The Linux specialist has also worked several years in the centralized IT, in Linux group. The group is responsible for Linux workstations in the university. In the VCS project, he is responsible for setting up the server environment for GitLab.

Two customer service persons have both been working in the university's IT Service Desk for more than four years. They both have extensive knowledge about university's IT systems. In the VCS project, they are responsible for user interviews, the market survey, the mapping of information security, creating the user instructions and edifying the system to the rest of the service desk.

The customer service people made an exception to the criterion since they have a little experience from project working. For them, the project was an opportunity to learn project working methods in practice and the project office can utilize them in future projects. This method as a way of learning was also highlighted by Pirinen when he said that learning usually takes place in groups and as a part of the work community (Pirinen, 2013, p. 14). The administrator of HIIT brought the customer view to the project. Though he had solid experience on working with version control systems before the project started, he would also provide valuable information on the user experience from the systems used in their department before this project.

3.2.2 Product comparison and feasibility study

The product comparison was made to find out what VSC's would be the most suitable for the university's IT environment. The products that were compared had to fulfill some predefined requirements, such as system features, system support, acquisition process, the possibility to customize the product and the price. "Technologies are often developed in response to specific task requirements using practical reasoning and experiential knowledge." (Hevner & Chatterjee, 2010, p. 15.)

The products that were chosen for the comparison were discovered by searching the internet. Also, most of the systems already used at the university were included in the comparison. The system features that were examined were authentication methods, workflow management systems, and server management, the number of repositories and simultaneous users and possibility to integrate the product with other systems. The support of the system, including possible updates, was considered a necessary feature. University's IT environment is constantly changing and the new revision control system needs to work in all situations, with the system that are used currently and the system which will be introduced in future.

The Acquisition process needed to fulfill the requirements of the university's procurement rules which are widely based on the Finnish and EU laws of procurements. The university has a set of rules that need to be followed when purchasing a product or service.

The product needed to be customizable to suit better the needs of the university. The possibility to widely customize the system settings and to integrate 3rd party plugins to the system. A highly customizable system also helps the university to integrate the product with other systems used at the university.

The price needed to fit into the university's annual budget. The cost of the product, support and update costs and system expansions costs were also examined as a part of the comparison. The VCS that fulfilled these requirements were selected for feasibility study.

3.2.3 Comparison process

During the product comparison, a selection of VCS products was compared with the pre-requirements (see appendix 1 for the requirements) determined by the project group. Though all the products were based on GIT technology, one of the requirements was compatibility with SVN-technology.

The older and nowadays more seldom used SVN technology was also included in the comparison though during the case study it was revealed to researchers that SVN was used only by a few users. The compared VCS products were chosen (see appendix 2 for the comparison table) based on the number of their users and on the release date and update history of the system. Old systems, without recent updates and small number of users were excluded from the comparison.

In our experience, the popularity of the system often guarantees that the system will be running in the future and it will be updated regularly. The comparison process was shown in Chapter 2.2.4, in Figure 6. The information was mostly gathered from the Internet, but literature about different VCS were also used to back up the data gathered from internet.

After the data were gathered, it was combined and filtered, in order to get only necessary data. After that the comparison was made the results were documented. The combined data about different version control systems was analyzed by comparing it with the pre-requirements. Based on the results of the analysis the version control systems that were closest to the pre-requirements were chosen for a feasibility study. The pre-requirements set several features, that the possible new version control system should fulfil. The features that were compared were the systems repository model, LDAP support, SVN support, supported platforms and cost.

The new system should have distributed, client-server repository, LDAP and SVN support and it should run on as many platforms as possible and be as economical as possible. In the end, the system that fulfilled the pre-requirements best, was GitLab Community Edition (later abbreviated as GitLab CE).

Gitlab CE has a built-in LDAP support, so the system can be configured to use Aalto's Active Directory for authentication the authentication question had the second most weight in requirements, as the primary requirement was the price of the future software.

Aalto's security policies had an impact on the comparison results as well. If the data would be hosted on a cloud-server, the server should be in EU region. During the initial setup of the

GitLab CE for Aalto University, it was decided that the system itself and the data stored in the system are hosted in the university's own premises.

3.2.4 Feasibility study

After the product comparison, a feasibility study was conducted. The main goals for the study was to find out if any of the selected products would suit the university's needs from the technical, economical and operational point of view. The phases of the feasibility study as seen by Rodney Overton(Overton, 2007, p. 6) adapted into the version control system acquisition project.

1. Technical feasibility told whether the suggested VCS could be supported with the existing technology of the IT department.
2. Economic feasibility told if the suggested VCS was cost effective.
3. Operational feasibility told would the VCS work in the organization if implemented.

The feasibility study of the project group found several suitable VCS systems. In case that the study would not have found a suitable system, the project would have had to rethink the requirements for the new VCS system. "A Feasibility Study can assist with making the crucial decision of whether to proceed or not in a wide range of business issues" (Overton, 2007, p. 6). During the feasibility process, it became apparent that HIIT was also planning to change their version control system and had already conducted similar studies, such as a market survey.

HIIT also had very similar requirements for the system and the project utilized their market survey. HIIT had concluded that GitLab would suit their needs. After a short analysis, the project group came to the same conclusion. GitLab would fulfill the requirements of the feasibility study.

Since many of the university's departments use GIT, the repositories would be easy to transfer to a similar system. HIIT had also come up with a solution how to synchronize SNV repositories. This would solve the problem how to transfer SVN repositories into GIT environment. GitLab also fulfilled the rest of the requirements for the system.

There are two editions of the GitLab, enterprise and community edition and the project group chose the community edition. When compared with the enterprise edition the community has less features, but it also costs significantly less.

3.2.5 Answering the research question

The research question for the second study was "What are the preparations prior to a revision control project". Finding the answer to the research question after the project had started, was quite a linear task. First, a preliminary report on the needs of a new system was composed, in addition, the preliminary report was complemented with customer survey results. Then the schedule of the project was created, then a project group was formed, after the project group was established, product comparison and feasibility study was conducted. After the studies were completed an acquisition proposal was made and when the SGPP approved the acquisition, the project was ready to start.

Most of the tasks done during the preparations could have also been done during the project itself but in order to save time used to the project, jobs that could be done prior the project itself, was made before the actual project kick-off. In this way, the project itself will allot much less resources.

Based on the preparation process, one should maybe rephrase the research question to a something more specific that would benefit the future project even more than simply reciting the task done in this project in hand. For example, how could the preparations process be optimized in order to gain the full advantage of the preparations during the project itself?

Optimization could be done by assigning certain people to complete the same steps prior to each process, currently the author of tasks, which recur in different projects, are nearly every time completed by a different person. The con of this method is that if the person has not done a certain task before, he must familiarize himself with the task, rather than if the same person would complete the same task in multiple projects, a certain routine would be achieved and therefore time would be spent much less. The findings of the preparation study have been compiled into Table 6.

Table 6 shows the multiple stages of the preparations phase with the results and benefits of each stage. They are listed in chronological order. The table presents the results and advantages of each stage either for the preparation stage, or the project itself, in simplified form.

Preparation task	Results	Benefits for the project
Forming of the project group	Workers with time, knowledge and intensity will actualize the project	The project has workers who will actualize it
Definitions	Project workers will know and understand the basics of VCS	The project workers will understand the context of the project
Timetables	The project will have an assessment about how much time each project task will take	Possibility to estimate how long the whole project will take
Surveys / User interviews	Knowledge about the current situation	The project will have data about what the new system should have
Market Survey	List of VCS that fulfill most or all the pre-requirements	Information about the most common VCS' on the market
Product comparison	Information about each examined VCS's features and functions	Knowledge about the possible "plan b", in case the original VCS isn't suitable after all
Feasibility study	Suitability Information of different examined systems from technical, economical and operational point of view.	Most suitable system will be acquired and installed
Project Plan	Rudimentary plan for the tasks and progression of the project	The project will advance in organized manner
Suggestion of investment for the university	Green light for the actualization of the project	The project will be actualized, instead of shutdown
Risk analysis of the project	Risks that threaten the project are recognized	The project can be concluded without any disruptions
Communication plan for the project	Project will use agreed communication tools	The project won't have information disconnections
Assignments	Each member had several task assignments	All the project tasks will be accomplished

Table 6: Compilation of tasks, results and benefits

By interpreting the results and benefits of each stage, a conclusion was made, that few of the stages could have been avoided by combining some of the stages.

Market survey and product comparison could have been one entity, where comparison could have been done simultaneously, when the available programs were charted. As both stages addressed practically the same subject, it would be reasonable to complete them at the same time to avoid the unnecessary use of time by, for example, documenting both stages one by one when only one documentation would have been enough.

All these tasks are not compulsory, but in this case, they were carried out. The essential tasks for similar projects should be chosen based on the experience and the knowledge of the project members. Experienced project group could run the preparations with lesser number of tasks and faster pace.

3.3 Study III: Testing the version control system

The aim of the 3rd study is to provide important information about testing during a project to Aalto University's centrally managed IT and an effective way for them to conduct tests during different projects in the future.

This study is based on the testing conducted in Aalto University during a VCS project which was started in 2015. In addition, using the data compiled during the project, also literature and publications about software testing will be compared with the research results. The research questions in the 3rd study were following:

1. How should the testing be performed during a VCS project?
2. How can the test results be used to improve the user experience?

By finding answers to the research questions, will improve the testing conducted by centralized IT and knowledge about how to exploit the test results to supply even better user experience for the end-users of the system.

3.3.1 Project and testing adapted to waterfall model

At the beginning of the project, a need for a centrally managed version control system was discovered based on the user interviews (Miettinen & Nousiainen, 2015). The design for the future system was planned after a project group was established, project group consisted of people, who had prior knowledge of the future system and the way the system should work.

After the project group was established, certain areas of responsibilities were distributed among the group, the more technical people were responsible for building the environment and less experienced people focused on marketing plans and planning the training requirements. During the verification phase, a group of pre-planned tests were performed and the test results were carefully documented. The last step of waterfall model, maintenance step will encase monthly administrative tasks, such as version updates for the system and possible feature updates which has been received from the users of the system. The iterative waterfall model is described in Figure 7.

3.3.2 Testing

What is testing and why is it done? What can be achieved when an already released program is tested by the company that is integrating a new software into its repertory?

The art of software testing (Myers, Sandler, & Badgett, 2011, p. 6) defines testing as a way of finding errors in the program while it is used. The definition is quite accurate, but in Aalto's case, one could also describe testing as a method of enhancing the future user experience by implementing certain functions into the software which are not automatically included in it when the software is the first time installed. Such features were related to the login, and administrating the repositories.

3.3.3 Planning the tests

During the planning phase, it is a required custom to create a certain number of test cases which are documented during the planning. In the project concerned, the test plan had various test cases which spanned from testing security to maintenance and from usability to accessibility.

Most of the test cases were planned so that they could be carried on by a user who has never used the system which was to be tested. Before the planning of test cases started, certain required fields for the test had to be determined. The testing was done by following pluralistic methods introduced by Jakob Nielsen in his study (Nielsen, 1994, p. 413). The fields in this project are presented in Table 7:

Field	**Content of the field**
Test area	Which area of system were tested.
Features	Specific area or function which was bee tested.
Test case description	Short description of the test case, what was tested in the scenario.

Setup	What are the pre-requirements before the test could be conducted?
Execution	A more specific, step by step instructions how the test should be conducted.
Expected result	What was expected to happen during the test scenario, how the test was supposed to end.
Ending	Where does the specific test scenario end?
Date performed	When the test was performed, information is vital to determine the version of software which was in production during the test scenario.
Result	The person who conducted the test scenario will fill out observations done during the test scenario.
Comments	Observations about the scenario, possible development proposals to the technical team.
Name of the tester	The person who performs the test will mark his or her name to the field, the name will help the people who are responsible for the program development, to contact the tester if they require some additional information about the results.

Table 7: Fields used in the testing document

3.3.4 Conducting the tests

The test cases were conducted by the project group as no one of the personnel who works for Aalto's central IT, has specialized in testing. The project group consisted of six people, three of them had previous knowledge of VCS systems and the way the system should work.

As mentioned before, each test scenario was carried out by each member of the project group, so the testing group consisted from only six people. The project manager of the VCS project was very confident that the results of the testing will be sufficient though the test group was quite small. At this point, it also wise to mention that in Aalto's VCS project, every other area than security, will be tested manually without any software that would automate the tests. The vulnerability assessments for the software will be scanned with an automated scanner provided by a company called Tenable Network Security, Inc.Conducting the tests

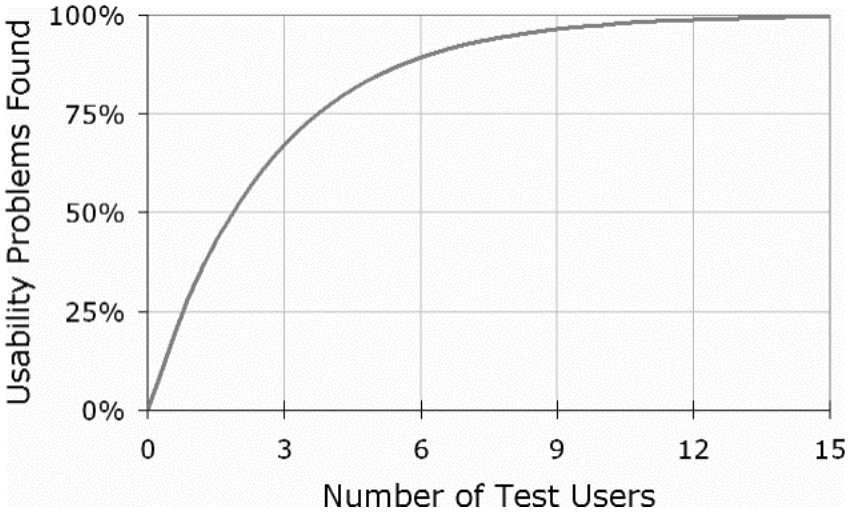

Figure 6: Problems found vs. the number of testers (Nielsen, March, 2000)

Jakob Nielsen's article backs up the small size of the test group and Mr. Nielsen also states in his article the following: "The best results come from testing no more than five users and running as many small tests as you can afford." (Nielsen, March, 2000) Nielsen also explains why increasing the number of testers is only basically waste of resources and there is not that much to gain when you are using more than five testers. As Figure 6 shows, over 80% of the usability problems can be found with only five testers.

As the testers in Aalto University's VCS project were the same people who were responsible for setting up, updating and developing the system, the problems noticed during the tests were solved also during the tests. Nielsen also backs up this method in his article by writing following "After the first study with five participants has found 85% of the usability problems, you will want to fix these problems in a redesign" (Nielsen, March, 2000).

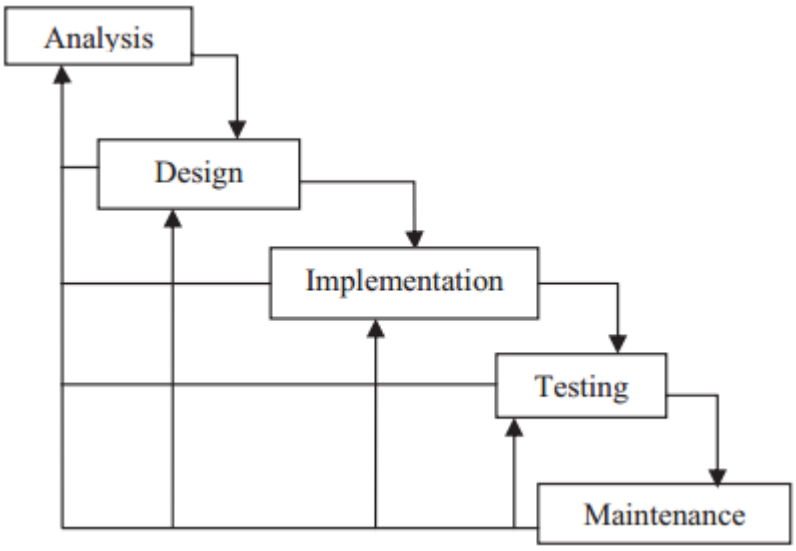

Figure 7: Iterative waterfall Software Development Model (Trivedi & Sharma, 2013)

The testing phase in VCS project could be compared with the iterative waterfall software development model presented in Figure 7. The issues discovered during the tests could be restored back to the design – stage, and after that the feature could be patched and then implemented back to the system and tested again. In some test cases, the result of the test could even undergo the analysis again to determine, will the feature be implemented into the system or could it be replaced with another similar feature, such feature being one of the login features in the system.

3.3.5 Reporting the test results

" Always use a certain form to gather a certain kind of information." (Hutcheson, 2003, p. 20.)Test reports can be considered as a vital information for the people who develop the systems. The more rigorously the test results are documented, the easier the results can be used to patch the issues found during the tests. The most practical way to report the test results is to store them to the same document where the test cases are created during the planning phase were documented.

In Aalto University's VCS project, the test scenarios were documented in an Excel-sheet and after a scenario was completed, the tester would document the results as accurately as possible to the same Excel-sheet. An example of the test document can be seen in Figure 8. After the tests were done, the sheets were gathered and results were analyzed. Based on the analyzation, required action was taken and possible issues were patched.

Test area	Feature	Test case description	Setup	Execution	Excpected results	Ending	Date	Result	Comments	Tester
Accessibility	OS Test / Platform	Access from Linux will be tested	Access to the system from Terminal will be tested				19.9.2016	Passed	Test repository was cloned on the desktop without errors	
Accessibility	OS Test / Platform	Access from Windows will be tested	Install a git-client (https://desktop.github.com/) on a Windows computer	Git-client will be installed on an Windows workstation and after that access to the system with the client will be tested		User is able to connect to a repository with the client installed on the system.	19.9.2016	Passed	Test repository was cloned on the desktop without errors	Jani
Accessibility	OS Test / Platform	Access from OSX will be tested	Install a git-client (https://desktop.github.com/) on a OSX based system	Git-client will be installed on an OSX workstation and after that access to the system with the client will be tested		User is able to connect to a repository with the client installed on the system.	19.9.2016	Passed	Test repository was cloned on the desktop without errors	Jani

Figure 8: Example of the test document

3.3.6 Answering the research question

As mentioned in this study, testing is an essential part of enhancing the user experience. Though every bug cannot be found while conducting the tests, most of them can be removed during and after the tests. When bugs are removed, the experience for the user is more fluent, a fluent work platform encourages users to use the new system and transfer their data from the old systems to the new, centrally managed VCS system.

When it comes to answering the research questions, one could say, that according to the literature and researches published about testing, there is no correct way to conduct the tests in a VCS project, rather the testing should be a hybrid of many different test methods. One could also mention, the test methods only provide a guideline how the tests could be conducted, but it is always up to the project manager to decide how the tests are conducted in different projects.

The secondary research question was already answered in this study earlier. The better the results are documented; easier they can be used to patch the issues found during the tests. The easier the issues can be introduced to the system, the better the user experience will be.

4 Results

The research process followed the process flowchart introduced in Figure 9. This study was based on four of the five separate steps which all were done during the project. Each study had different methodology and the methodology used in these studies were compared with the studies released prior.

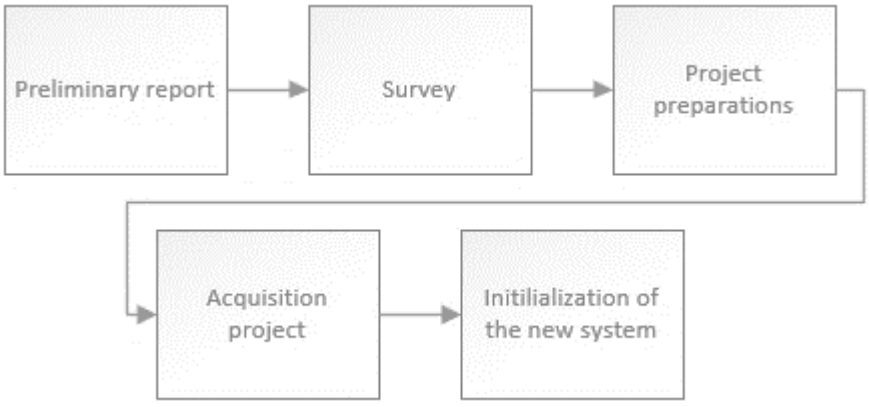

Figure 9: Flowchart of the processes in the project

Results of each study were presented at the end of each study. To summarize the results and answer the research question of the whole study, the results will be reviewed in this chapter once more.

The question for the first study aimed to find out, how the usage of version control system could be improved at Aalto University?

By reducing the overlapping systems, users and support personnel can familiarize themselves with a single system and administrators can focus on developing a single system instead of maintaining several systems and trying to introduce new features in between maintenance jobs. The second study debated on the project preparations. The research question aimed to clarify the tasks which had been completed prior to the version control system acquisition project in hand. Study showed that multiple tasks were completed before the actual project was even started. The study also spoke out how the processes could be optimized to save time and resources during the actual project.

Third study focused on the testing conducted during the project itself. The study debated on how the testing should be done and how the test results can be used to improve customer experience on the new system. Test results offer a solid base for the system development, to enhance the customer experience, the system should be modified per the test results. Though the project group was small, testing could be done effectively, some reports even state that using over 5 testers on a system can be found as the waste of resources (Nielsen, March, 2000).

Study	**Question of the study**	**Results**
Customer demand Survey	How could the usage of version control systems be improved in Aalto University?	Amount of systems should be reduced in order to make the usage more efficient
Preparation tasks	What are the preparations prior a version control system project?	Essential preparation tasks were identified. Tasks, which could be avoided were identified
Testing the version control system	How should the testing be performed during a VCS project? How can the test results be used to improve the user experience?	Testing requires no more than 5 persons. Testing should be accurately documented Results can be used to technically enhance the system to meet the customer demands

Table 8: Summary of studies

4.1 Main theme of the study

The main theme of this study was "How to comprehend, rationalize and develop version control systems?"

The three studies conducted in a different way helped the researcher answer the main theme. The version control system in Aalto University can be comprehended as a strong collaboration tool for both, students and staff. When the system is used effectively, both parties will benefit from the usage, students will learn the usage of a system which is highly valued among the future employers, staff can use the system with external parties in common projects.

How could the system be developed and rationalized? The first step of rationalizing the system has already been done, as many departments which are currently hosting their own systems, will start using the centrally managed system and the amount of overlapping systems will deduct.

The development of usage of the version control system has also been launched, as a plain version control system could serve as a course platform in the near future. To serve as a course platform, minor changes to the newly acquired system must be done, but if the changes are justified, there is no reason why they would not be made and the system could serve as a course platform. In order to simulate real software development, where the VCS is an essential tool, a project management system could be introduced alongside the VCS in courses (Biňas, 2013, p. 23).

4.2 Implications and future research

This study offers a solid foundation for future studies. The methodologies used in the study are described in such fashion, that the can be adapted to either similar projects, or applied in completely different projects.

In future, this study could be a foundation for a study, where research would focus on optimizing the processes during a similar project. Besides optimizing the processes, study could also focus on the people related to the project, in the results of study 2, in paragraph 3.2.5 it was mentioned that the research question used in this study could be rephrased and studied from a different perspective. By studying the impact of person's experience on certain tasks, the aspect of optimization could be studied at the same time.

This study also articulates that by reducing the overlapping systems, resources can be saved and re-focused. Future studies could concentrate on the alleged resource savings and how the saved resources can be channeled to maximize the future development of the system acquired.

The results of this study were used in a conference paper "Version Control Systems - development, comprehension and rationalization of usage: Case Aalto University". It will be presented at the 5th International Conference on Management, Marketing, Tourism, Retail, Finance and Computer Applications (MATREFC '17). The abstract of that paper can be found in Appendix 3 (Appendix 3).

4.3 Replication of the study

This study can be replicated on the literal-level, however, as the environment, where the study was composed is a closed one, specific technical data about the environment and set up cannot be shared in this study. "Qualitative researchers, in contrast, typically use criterion sampling. This means that participants are selected based on a set of prescribed criteria established by researchers. Participants possess the variables of interest for study and may or may not represent others in the population from which the sample was drawn. Due to this choice of sampling methodology, qualitative researchers likely will always have a weak case for external validity when focusing solely on one individual study." (Given, 2008, p. 754.)

The fact that the interviewees were picked by the project group and local administrators may cause issues with the possible replication attempts as part of the study data solely rely on the opinions of the users of current systems. The replication of the study can be confrontational as it is highly unlikely that the observed configurations, people, social structures or programs are the same (Lee, 1989, p. 40).

5 Conclusion and discussion

The subject of the study is very topical as in future, numerous companies will find their selves in a similar situation where functions must be re-examined in order to find an edge in the field of fierce competition they are operating on. In this scenario, the little things can make the difference in between the survival of the company or ceasing to exists. The current state of economics forces companies to seek savings in their budgets, and in this scenario, overlapping systems are one of the most likely target of saving resources which lead the companies to save money when they do not have to administrate multiple systems at the same time.

By rationalizing the operational environment, both, universities and non-governmental companies enable their workers to focus more on their primary duties as the workers do not have to use several different systems to complete their daily tasks.

On the administrative side, the rationalization of operative environment means at least few different things. Instead of investing the limited work time into administrating multiple systems, work can be done more effectively and the time saved from focusing functions on fewer

systems, can be focused on the development of current systems or searching for a substituent for systems that still need to be replaced in order to acquire cost savings.

Based on this study, and the observations made during the project, it is fair to say that if possible, once a project has been completed successfully, the same people would be used to another project. As the people gain experience on project working, they will work faster and more efficiently as they already know what is expected of them and how the goals of the project are achieved. The role of the project manager is vital in an acquisition project. Manager is the one person, who will determine the schedule, and it is up to administrators to meet the schedule the manager has determined. Though the project manager is the one who composes the schedule, it is important that he consults with the administrators before drawing up the schedule for a project, as the administrators may be involved in several projects at the same time, this will allow the manager to compose a realistic schedule and then the schedule will not have to be re-adjusted during the project itself.

Based on the study observations, it is fair to say that when you are planning a project that will affect directly the end-users, if possible, include the view of end-users in the project plan before starting the project itself. In this study, the user interviews compiled before the project started, offered valuable information about the usage of current systems and about the expectations for the future system. Without this information, a great system can be built, but it is a completely different thing if the end-users are willing to use the new system as it does not meet their expectations.

5.1 Ethics and reliability of the research

In qualitative research "Qualitative researchers focus their research on exploring, examining, and describing people and their natural environments. Embedded in qualitative research are the concepts of relationships and power between researchers and participants. The desire to participate in a research study depends upon a participant's willingness to share his or her experience" (Orb, Eisenhauer, & Wynaden, 2001, p. 93). In this study the willingness of the interviewees was guaranteed by asking their permission to record the interview sessions and by promising them that the audio files will be deleted after the transcription. The interviewees were also promised a total anonymity and their identities were hidden even from the management of the project.

"Reliability, in the field of research, is broadly described as the dependability, consistency, and/or repeatability of a project's data collection, interpretation, and/or analysis." (Given, 2008, p. 753.) In this study, reliability was ensured with dependability and repeatability. Dependability of qualitative research investigates if the researcher has made any mistakes during the data collection process, when interpreting the data or when reporting the data. These were

avoided by using a dependability audit in where an outside person with knowledge about the subject of the study analyses the actions of the researchers and gives feedback. In this study, there were two auditors. One was from the university and the other was from outside of the university.

Repeatability in qualitative research investigates what kind of results would the study give if it would be done again and how similar would the results be with the original results. The study can be partially repeated. As the data gathering relied partially on information acquired with semi-structured interviews, results can differ from results presented in this study.

Repeatability of the study is possible since all the research material used in study is available and the interviewees are also available.

5.2 Limitations

Though the end results corresponded the hypothesis of the study, the results might have been different. Foremost, the sampling of the interviews was rather small and all the people interviewed, had solid knowledge on the usage of version control systems. If the interview sampling would have been bigger, common attributes identified with the interviews could have been very different. Different attributes could have altered the processes required during the preparations and the actual project.

Secondly, this was the first time when both researchers participated in an acquisition project from the very beginning. Though the research relied heavily on literature and studies released about the tasks conducted during the project, the in-experience towards project working might reflect in the research results though every study result was interpreted as subjectively as possible by reflecting the result on similar researches. As few minor limitations were discovered when reflecting the whole project, the results of the research can still be considered accurate.

References

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly,* , 369-386.

Biňas, M. (2013). Version control system in CS1 course: Practical experience. *IEEE International Conference on Emerging eLearning Technologies and Applications (ICETA), , 11* 23-28.

Chacon, S., & Straub, B. (2014). *Pro git* Apress.

Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches* Sage publications.

Davenport, T. H., & Beers, M. C. (1995). Managing information about processes. *Journal of Management Information Systems, 12*(1), 57-80.

De Leeuw, E. D., & Dillman, D. A. (2008). *International handbook of survey methodology* Taylor & Francis.

Didcoe R., S. B. (July, 2007). *Feasibility study guide.* ().Department of Sport and Recreation. Retrieved from http://www.dsr.wa.gov.au/support-and-advice/facility-management/developing-facilities/feasibility-study

Dubé, L., & Paré, G. (2003). Rigor in information systems positivist case research: Current practices, trends, and recommendations. *MIS Quarterly, 27*(4), 597-636.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14*(4), 532-550.

Fink, A. S. (2000). The role of the researcher in the qualitative research process. A potential barrier to archiving qualitative data. *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research, , 1*(3)

Folkestad, B. (2008). *Analysing interview data possibilities and challenges* (no. 13). *Eurosphere Working Paper Series. Online Working Paper,*

Given, L. M. (2008). *The sage encyclopedia of qualitative research methods* Sage Publications.

Guest, G., Namey, E. E., & Mitchell, M. L. (2012). *Collecting qualitative data: A field manual for applied research* Sage.

Haaranen, L., & Lehtinen, T. (2015). Teaching git on the side: Version control system as a course platform. *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education,* 87-92.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian Journal of Information Systems, 19*(2), 87-92.

Hevner, A., & Chatterjee, S. (2010). Design research in information systems: Theory and practice.

Hutcheson, M. L. (2003). *Software testing fundamentals: Methods and metrics* John Wiley & Sons.

Kamberelis, G., & Dimitriadis, G. (2013). *Focus groups* Routledge.

Kaplan, B., & Duchon, D. (1988). Combining qualitative and quantitative methods in information systems research: A case study. *MIS Quarterly,* , 571-586.

Kaplan, B., & Maxwell, J. A. (2005). Qualitative research methods for evaluating computer information systems. *Evaluating the Organizational Impact of Healthcare Information Systems,* , 30-55.

Kelleher, J. (2014). Employing git in the classroom. *Computer Applications and Information Systems (WCCAIS), 2014 World Congress On,* 1-4.

Kertész, C. (2015). Using GitHub in the classroom-a collaborative learning experience. *Design and Technology in Electronic Packaging (SIITME), 2015 IEEE 21st International Symposium For,* 381-386.

Kitzinger, J., & Barbour, R. (1999). *Developing focus group research: Politics, theory and practice* Sage.

Lähteenmäki, I. (2015). *Preliminary report on version control system.* (Internal report). Espoo: Aalto University.

Lee, A. S. (1989). A scientific methodology for MIS case studies. *MIS Quarterly,* , 33-50.

Leffingwell, D. (2010). *Agile software requirements: Lean requirements practices for teams, programs, and the enterprise* Addison-Wesley Professional.

Merriam, S. B., & Tisdell, E. J. (2015). *Qualitative research: A guide to design and implementation* John Wiley & Sons.

Miettinen, A., & Nousiainen, J. (2015). *How could the usage of revision control systems be improved in university X.* (Case study). Espoo:

Mitchell, M. L., & Jolley, J. M. (2012). *Research design explained* Cengage Learning.

Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* John Wiley & Sons.

Nielsen, J. (1994). Usability inspection methods. *Conference Companion on Human Factors in Computing Systems,* 413-414.

Nielsen, J. (March, 2000). Why you only need to test with 5 users. Retrieved from https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/

Nunamaker Jr, J. F., Chen, M., & Purdin, T. D. (1990). Systems development in information systems research. *Journal of Management Information Systems, 7*(3), 89-106.

O'Grady, J. V., & O'Grady, K. (2009). *A designer's research manual: Succeed in design by knowing your clients and what they really need* Rockport Publishers.

Orb, A., Eisenhauer, L., & Wynaden, D. (2001). Ethics in qualitative research. *Journal of Nursing Scholarship, 33*(1), 93-96.

Overton, R. (2007). *Feasibility studies made simple* Martin Books.

Pirinen, R. (2013). Towards realization of research and development in a university of applied sciences. *Publications of the University of Eastern Finland, 108*

Powell, R. R. (2006). Evaluation research: An overview. *Library Trends, 55*(1), 102-120.

Ravitch, S. M., & Carl, N. M. (2015). *Qualitative research: Bridging the conceptual, theoretical, and methodological* SAGE Publications.

Raymond, E. (2007). Understanding version-control systems (draft). Retrieved from http://www.catb.org/esr/writings/version-control/version-control.html

Reilly, R. C. (2010). Participatory case study., 658-660.

Rochkind, M. J. (1975). The source code control system. *IEEE Transactions on Software Engineering,* (4), 364-370.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering, 14*(2), 131-164.

Ruparelia, N. B. (2010). The history of version control. *ACM SIGSOFT Software Engineering Notes, 35*(1), 5-9.

Singh, Y. K. (2006). *Fundamental of research methodology and statistics* New Age International.

Sink, E. (2011). *Version control by example* Pyrenean Gold Press.

Thomas, D. R. (2006). A general inductive approach for analyzing qualitative evaluation data. *American Journal of Evaluation, 27*(2), 237-246.

Trivedi, P., & Sharma, A. (2013). A comparative study between iterative waterfall and incremental software development life cycle model for optimizing the resources using computer simulation. *Information Management in the Knowledge Economy (IMKE), 2013 2nd International Conference On,* 188-194.

TutorialConflict. (2013). Tutorial - merging conflicting changes. Retrieved from https://www.mercurial-scm.org/wiki/TutorialConflict

Yin, R. K. (2009). *Case study research : Design and methods* (4th ed ed.). Thousand Oaks (Calif.): Sage Publications.

List of figures

List of tables

List of appendixes

Appendix 1: List of requirements

Functional requirements

- Control of user rights for a single repository, which the end-user can manage
- Private repositories and a possibility to use shared repositories
- Git push notifications
- Export / download the repository as a zip-file
- Easy to maintenance and manage
- Controlled lifespan of repositories and credentials (for administration)
- Easy conversion / migration to the new repository keeping the history (a tool for transferring old Git/SVN repositories to the new system)
- Hooks for Push/Pull functions
- No limitations for the number of active projects
- Possibility to change the ownership of projects
- Responsive support to increase project limit

Use and usability

- User friendly
- Support for all device platforms
- Support for most common development environments
- Possibility to use Aalto's credentials, SSO, username/password and SSH keys
- HAKA authentication
- Possibility to collaborate with people outside of the university (username and password possibility for outsiders)
- The students should be able to do group assignments with the new system and single student must be able to define the members of the group
- Access from the Internet
- Possibility to SSH & HTTPS Push / Pull from all the networks
- Product branding. The new VCS must be recognized as university's service

Form of data

- Text based data, mostly programming code, also configuration files, latex files and so on
- Binary files (pictures, PDF, docx) only infrequently

Information security

- The system needs only basic level information security (ST IV, internal). It can therefore be applied to a basic-level's list of security requirements
- See the separate excel file about information security requirements

Backups

- Backups are needed. The VCS system will be backed up in a similar way that file systems or virtual machines are backed up.
- The size of the code is gigabytes (possible binary files are big). The size of all code (all the versions of the code) of Aalto should fit into 500 gigabytes.

Data protection and privacy

- Only the relevant information for usage is stored (for example username, email address, organization)
- Possibility to separate Aalto's users from other users
- "Hidden mode" – the repository doesn't show in any listings, access only if the person knows direct address to the repository, the repository won't be indexed
- Possibility to change privacy settings of a single repository at any time during its' lifespan
- Rules of HAKA (the requirements that all the services attached to HAKA must fulfill)

Data Content and confidentiality

- In most cases the information isn't public, but not entirely confidential or delicate. In which case, basic level information security (ST IV, internal) fulfills the requirements
- In some projects enterprise and business secrets require case-by-case negotiations, if Aalto's usual solution isn't suitable, it might be best to use a separate environment (or environment offered by the company). These have been estimated at a minimum
- Configurations are internal and not necessary to include to the VCS system. Configurations can be kept in a different service, which can only be accessed from the intranet

Availability

- 99% availability is not required; the system isn't critical and can be offline for few moments
- Short outage doesn't cause damage for most of the user base (deadlines vary). GIT users have possibility to use local copy during an outage
- For most people of the user base a short outage doesn't cause
- Monitoring? Possibility to inform the users about outage at webpage
- Users can be informed of a break e.g. Web page (cf. Wwwdown.aalto.fi)? At least the normal practices of error information.

Continuance

- Continuance will be ensured with maintenance documentation and by ensuring maintenance knowledge with deputies
- Reliability and integrity are important; information must remain unchanged in a safe place

Appendix 2: Product comparison table

Software	**Repository model**	**LDAP Support**	**SVN Support**	**Supported platforms**	**Cost**
GIT	Distributed	X		Unix, Windows, OSX	Free to use, Prices of host services vary
GitHub Enterprise	Client-Server	X	X	Unix, Windows, OSX	$2500 for team size of 10, $23,750.00 for team of 100
GitLab CE	Client-Server	X	X	Unix, Windows, OSX	Basic: $39/year, Standard: $49/year, Plus: $149/year. Prices are for a single-user licenses
Subversion	Client-Server		X	Unix, Windows, OSX	Free. Prices of host services vary
Mercurial SCM	Distributed	X	X	Unix, Windows, OSX	Free software licensed under the terms of the GNU General Public License Version 2 or any later version.

Appendix 3: Abstract and keywords for the conference paper “Version Control Systems - Development, Comprehension and Rationalization of usage: Case Aalto University”

Abstract: IT environments are a continuously changing entirety. Software and systems in these environments go through their lifecycles and then they are replaced with more advanced and economical solutions. This is a normal phenomenon in education and corporate life. The study examines if and how the different systems could be replaced with a single version control system (VCS) and how to prepare and execute an acquisition project of a new IT system. The research method is a multiple case study composed of three case studies. The findings show that users are ready to change their VCS tools to new ones, as long as they get to manage their own repositories. The departments are ready to renounce their own systems as long as the new system has all the same functions as the old systems. The reduction of the overlapping systems will also save money and the resources of the departments.

Keywords: Case study; Version control system; Acquisition; Rationalization