Aalto University
School of Electrical Engineering
Degree Programme in Automation and Systems Technology

Sakari A. Pesonen

An open and general CNC and machine vision based architecture for payment terminal acceptance testing automation

Master's Thesis
Espoo, May 13, 2016

DRAFT! — April 24, 2016 — DRAFT!

Supervisor: D.Sc. Seppo Sierla, Aalto University
Advisor: Tatu Kairi M.Sc.

Aalto University
School of Electrical Engineering
Degree Programme in Automation and Systems Technology

ABSTRACT OF
MASTER'S THESIS

Author:	Sakari A. Pesonen		
Title:	An open and general CNC and machine vision based architecture for payment terminal acceptance testing automation		
Date:	May 13, 2016	**Pages:**	vi + 20
Major:	Intelligent Products	**Code:**	T-110
Supervisor:	D.Sc. Seppo Sierla		
Advisor:	Tatu Kairi M.Sc.		

A dissertation or thesis is a document submitted in support of candidature for a degree or professional qualification presenting the author's research and findings. In some countries/universities, the word thesis or a cognate is used as part of a bachelor's or master's course, while dissertation is normally applied to a doctorate, whilst, in others, the reverse is true.

!FIXME **Abstract text goes here (and this is an example how to use fixme).** FIXME! Fixme is a command that helps you identify parts of your thesis that still require some work. When compiled in the custom `mydraft` mode, text parts tagged with fixmes are shown in bold and with fixme tags around them. When compiled in normal mode, the fixme-tagged text is shown normally (without special formatting). The draft mode also causes the "Draft" text to appear on the front page, alongside with the document compilation date. The custom `mydraft` mode is selected by the `mydraft` option given for the package `aalto-thesis`, near the top of the `thesis-example.tex` file.

The thesis example file (`thesis-example.tex`), all the chapter content files (`1introduction.tex` and so on), and the Aalto style file (`aalto-thesis.sty`) are commented with explanations on how the Aalto thesis works. The files also contain some examples on how to customize various details of the thesis layout, and of course the example text works as an example in itself.

Keywords:	ocean, sea, marine, ocean mammal, marine mammal, whales, cetaceans, dolphins, porpoises
Language:	English

Aalto-yliopisto
Sähtkötekniikan korkeakoulu
Automaatio- ja systeemitekniikan koulutusohjelma

DIPLOMITYÖN
TIIVISTELMÄ

Tekijä:	Sakari A. Pesonen		
Työn nimi:	Ohjelmistoprosessit mänteille		
Päiväys:	13. toukokuuta 2016	**Sivumäärä:**	vi + 20
Pääaine:	Älykkäät tuotteet	**Koodi:**	T-110
Valvoja:	D.Sc. Seppo Sierla		
Ohjaaja:	Filosofian maisteri Tatu Kairi		

Kivi on materiaali, joka muodostuu mineraaleista ja luokitellaan mineraalisisältönsä mukaan. Kivet luokitellaan yleensä ne muodostaneiden prosessien mukaan magmakiviin, sedimenttikiviin ja metamorfisiin kiviin. Magmakivet ovat muodostuneet kiteytyneestä magmasta, sedimenttikivet vanhempien kivilajien rapautuessa ja muodostaessa iskostuneita yhdisteitä, metamorfiset kivet taas kun magma- ja sedimenttikivet joutuvat syvällä maan kuoressa lämpötilan ja kovan paineen alaiseksi.

Kivi on epäorgaaninen eli elottoman luonnon aine, mikä tarkoittaa ettei se sisällä hiiltä tai muita elollisen orgaanisen luonnon aineita. Niinpä kivestä tehdyt esineet säilyvät maaperässä tuhansien vuosien ajan mätänemättä. Kun orgaaninen materiaali jättää jälkensä kiveen, tulos tunnetaan nimellä fossiili.

Suomen peruskallio on suurimmaksi osaksi graniittia, gneissiä ja Kaakkois-Suomessa rapakiveä

Asiasanat:	AEL, aineistot, aitta, akustiikka, Alankomaat, aluerakentaminen, Anttolanhovi, Arcada, ArchiCad, arkki
Kieli:	Englanti

Acknowledgements

I wish to thank all students who use LaTeX for formatting their theses, because theses formatted with LaTeX are just so nice.

Thank you, and keep up the good work!

Espoo, May 13, 2016

Sakari A. Pesonen

Abbreviations and Acronyms

2k/4k/8k mode	COFDM operation modes
3GPP	3rd Generation Partnership Project
ESP	Encapsulating Security Payload; An IPsec security protocol
FLUTE	The File Delivery over Unidirectional Transport protocol
e.g.	for example (do not list here this kind of common acronymbs or abbreviations, but only those that are essential for understanding the content of your thesis.
note	Note also, that this list is not compulsory, and should be omitted if you have only few abbreviations

Contents

Chapter 1

Introduction

This is my master's thesis, and I am very proud of it. Of course, when I write my *real* master's thesis, I will not use the singular pronoun *I*, but rather try to avoid referring to myself and speak of the research *we* have conducted—I rarely work alone, after all. Yet, both *I* and *we* are correct, and it depends on the instructor and the supervisor (of course from you, too), which one they would prefer. Anyway, the tense should be active, and passive sentenses should be avoided (especially, writing sentences where the subject is presented with by preposition), so often you cannot avoid choosing between the pronouns. Life is strange, but there you have it.

By the way, the preferred order of writing your master's thesis is about the same as the outline of the thesis: you first discover your problem and write about that, then you find out what methods you should use and write about that. Then you do your implementation, and document that, and so on. However, the abstract and introduction are often easiest to write last. This is because these really cover the entire thesis, and there is no way you could know what to put in your abstract before you have actually done your implementation and evaluation. Rarely anyone write the thesis from the beginning to the end just one time, but the writing is more like process, where every piece of text is written at least twice. Be also prepared to delete your own text. In the first phase, you can hide it into comments that are started with % but during the writing, the many comments should be visible for your helpers, the instructor and supervisor.

The introduction in itself is rarely very long; two to five pages often suffice.

1.1 Problem statement

Undergraduate students studying technical subjects do not consider typography very interesting these days, and therefore the typographical quality of many theses is unacceptably low. We plan to rectify this situation somewhat by providing a decent-quality example thesis outline for students. We expect that the typographical quality of the master's theses will dramatically increase as the new thesis outline is taken into use.

1.2 Structure of the Thesis

You should use transition in your text, meaning that you should help the reader follow the thesis outline. Here, you tell what will be in each chapter of your thesis.

Chapter 2

Payment terminal acceptance testing

The problem must have some background, otherwise it is not interesting. You can explain the background here. Probably you should change the title to something that describes more the content of this chapter. Background consists of information that help other masters of the same degree program to understand the rest of the thesis.

Transitions mentioned in Section 1.2 are used also in the chapters and sections. For example, next in this chapter we tell how to use English language, how to find and refer to sources, and enlight different ways to include graphics in the thesis.

2.1 Benefits of Open Source solutions

Moreover, the transitions are also used in the paragraph and the sentence level meaning that all the text is linked together. For example, the word "moreover" here is one way, but of course you should use variation in the text. Examples of transitional devices (words) and their use can be found from writing guides, e.g. from the Online Writing Lab (OWL)[1] of Purdue University or Strunk's Elements of Style[2]. Remember that footnotes are additional information, and they are seldom used. If you refer to a source, you do no not use footnote. The right command for the references is *cite*.

The language used in the thesis should be technical (or scientifical). For example, the abreviations aren't used but all them are written open (i.e. "are not"). Since the content itself is often hard to understand (and explain),

[1] http://owl.english.purdue.edu/owl/resource/574/02/

[2] http://www.bartleby.com/141/

the sentences should not be very long, use complex language with several examples embedded in the same sentence, and, also, seldom used words and weird euphemism or paraphrases can make the sentence hard to follow and to read it with only one time, and making everything even harder to understand all this without any punctuation marks makes the instructor cry and finally after trying to correct the language, you will get boomerang, and everyone's time has just been wasted.

Please use proofreaders before sending even your unfinished version to the instructor and/or supervisor. You will get better comments when they do not need first proofread your text. Moreover, they can consentrate to the content better if the language and spelling mistakes are not distracting the reading. Several editors have their own proofreading tools, e.g. ispell in emacs. You can also use Microsoft Word to proofread your thesis: it can correct also some grammatical errors and not just misspelled words.

Note also that if you have a section or a subsection, you have to have at least two of them, or otherwise the section or subsection title is unnecessary. Same with the paragraphs an: you should not have sections with only one paragraphe, and single sentence paragraph. Furthermore, always write some text after the title before the next level title.

2.2 Common characteristics between payment terminals

Never ever copy anything into your theses from somebody else's text (nor your own previously published text). Never. Not even for starting point to be rewritten later. The risk is that you forgot the copied text to your thesis and end up to be accused of plagiarism. Plagiarism is a serious crime in studies and science and can ruin your career even its beginning. To repeate: never cut and paste text into your thesis!

2.3 Different approaches for test automation

All work is based on someone else's work. You should find the relevant sources of your field and choose the best of them. Also, you should refer to the original source where a fact has been mentioned first time. Remember source evaluation (criticism) with all sources you find.

Good starting points for finding references in computer science are:

- Nelli Portal (Aalto Library): `http://www.nelliportaali.fi`

- ACM Digital library: `http://portal.acm.org/`
- IEEExplore: `http://ieeexplore.ieee.org`
- ScienceDirect: `http://www.sciencedirect.com/`
- ... although Google Scholar (`http://scholar.google.com/`) will find links to most of the articles from the abovementioned sources, if you search from within the university network

Some of the publishers do not offer all the text of the articles freely, but the library has agreed on the rights to use the whole text. Thus, you should sometimes use computers in the domain of the university in order to get the full text. Sometimes the Nelli Portal can also help getting the whole article instead of just the abstract. The library has also brief instrucions how to find information Aalto University Library [2011b].

Instead of normal Google, use Google Scholar (`http://scholar.google.fi/`). It finds academic publications whereas normal Google find too much commercial advertisements or otherwise biased information. Wikipedia articles should be referred to in the master thesis only very, very seldomly. You can use Wikipedia for understanding some basics and finding more sources, but often you cannot be sure if the article is correct and unbiased.

One important part of the sources that you have found is the reference list. This way you can find the original sources that all the other research of the field refer. Often you can also find more information with the name of the researchers that are often referred in the articles.

2.4 Test suite syntax

The main point in referring to sources is to separate your own thinking and text from that of others. Facts of the research area can be given without reference, but otherwise you should refer to sources. This means two things: marking the source in the text where it has been used, and listing the sources usually in the end of the thesis in a way that help the reader to find the original source.

There are several bibliography styles, meaning how to form the bibliography in the end of the thesis. Aalto's library has good instructions for many styles Aalto University Library [2011a]. You should ask from your supervisor or instructors which style you should use. This thesis template uses the number style that is often used in software engineering. The other style also used in the CS field, e.g. usability, is the Harvard style where instead of numbers, the reference is marked into the text with author's name and

publishing year. Other areas use also many other styles for making the lists and marking the references.

In addition to the list in the end of the thesis, you have to mark the source in the text where the source is used. There are three places for the reference: in a sentence before the period, in the end of a sentence after the period, or in the end of a paragraph. All of them have different meaning. The main point is that first you paraphrase the source using your own words and then mark the source. Next, we give short examples that are marked with *emphasised text.*

Haapasalo Haapasalo [2010] researched database algorithms that allows use of previous versions of the content stored in the database. This kind of marking means that this paragraph (or until the next reference is given) is based on the source mentioned in the beginning. Giving the source you should use only the family name of the first author of the article, and not give any hints about what is the type of the article that is referred.

B+-trees offers one way to index data that is stored in to a database. Multiversion B+-trees (MVBT) offer also a way to restore the data from previous versions of the database. Concurrent MVBT allows many simultaneous updates to the database that is was not possible with MVBT. Haapasalo [2010] When the marking is after the period, the reference is retrospective: all the paragraph (or after previous reference marking) is based on the source given in its end. If the content is very broad, you can start with saying *According to Haapasalo*, then continue referring the source with several separate sentences, and in the end put the marking of your source *that shows that CMVBT are the best. Haapasalo [2010].*

If your paragraph has several sources, the above mentioned styles are not proper. The reader of your thesis cannot know which of your sources give which of the statements. In this case, it is better to use more finegraded refering where the reference markings that are embedded in the sentences. For example, *the multiversion B+-tree (MVBT) index of Becker et al. Becker et al. [1996] allows database users to query old versions of the database, but the index is not transactional. It's successor, the transactional MBVT (TMVBT), allows a single transaction running in its own thread or process to update the database concurrently with other transactions that only read the database Haapasalo et al. [2009]. Further development, titled the concurrent MBVT (CMVBT), allows several transactions to perform updates to the database at the same time Haapasalo [2010].* Here, the references are marked before the period in the sentences where they are used.

Finally, direct quotes are allowed. However, often you should avoid them since they do not usually fit in to your text very well. Using direct quotes has two tricks: quotation marks and the source. *"Even though deletions in*

a multiversion index must not physically delete the history of the data items, queries and range scans can become more efficient, if the leaf pages of the index structure are merged to retain optimality." Haapasalo [2010] Quotes are hard to make neatly since you should use only as much as needed without changing the text. Moreover, you often do not really understand what the author has mentioned with his wordings if you cannot write the same with your own words. Remember also that never cut and paste anything without marking the quotation marks right away, and in general, never cut and paste anything at all!

Sometimes getting the original source can be almost impossible. In an extremely desperate situation, you can refer with structure *mr X [...] according to ms Y [...] defined that*, if you find a source that refers to the original source. Note also that the reference marking is never used as sentence element (example of how **not** to do it: *Haapasalo [2010] describes an optimal algorithm for indexing multiversiond databases.*).

Chapter 3

Proposed architecture

A problem instance is rarely totally independent of its environment. Most often you need to describe the environment you work in, what limits there are and so on. This is a good place to do that. First we tell you about the LaTeX working environments and then is an example from an thesis written some years ago.

3.1 Overview

To create LaTeX documents you need two things: a LaTeX environment for compiling your documents and a text editor for writing them.

3.2 Hardware

You can write LaTeX documents with any text editor you like, but having syntax coloring options and such really helps a lot. My personal favourite for editing LaTeX is the *TeXlipse* Hupponen et al. [2010] plugin for the Eclipse IDE The Eclipse Foundation [2011]. Eclipse is an open-source integrated development environment (IDE) initially created for writing Java code, but it currently has support for editing languages such as C, C++, JavaScript, XML, HTML, and many more. The TeXlipse plugin allows you to edit and compile LaTeX documents directly in Eclipse, and compilation errors and warnings are shown in the Eclipse *Problems* dialog so that you can locate and fix the issues easily. The plugin also supports reference traversal so that you can locate the source line where a label or a citation is defined.

Eclipse is an entire development environment, so it may feel a bit heavy-weight for editing a document. If you are looking for a more light-weight option, check out TeXworks. TeXworks is a LaTeX editor that is packaged

with the newer MiKTeX distributions, and it can be acquired from `http://www.tug.org/texworks/`.

And if you are attached to your *emacs* or *vim* editor, you can of course edit your LaTeX documents with them. Emacs at least has syntax coloring and you can compile your document with a key binding, so this may be a good option if you prefer working with the standard Linux text editors.

3.3 Software

When you use `pdflatex` to render your thesis, you can include PDF images directly, as shown by Figure 3.1 below.

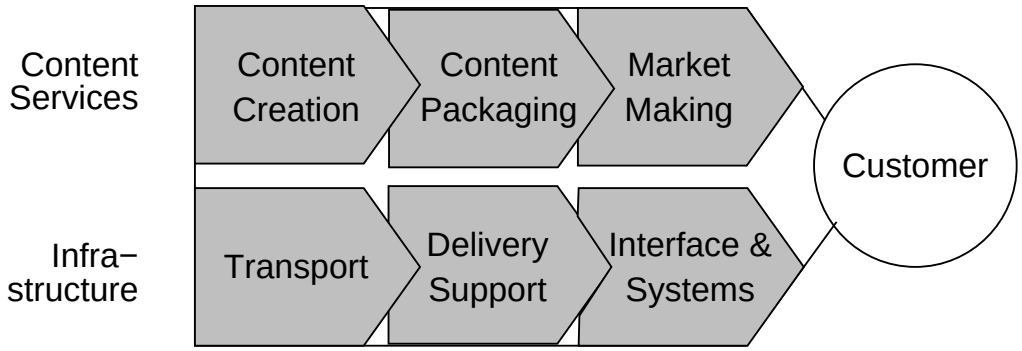


Figure 3.1: The INDICA two-layered value chain model.

You can also include JPEG or PNG files, as shown by Figure 3.2.

You can create PDF files out of practically anything. In Windows, you can download PrimoPDF or CutePDF (or some such) and install a printing driver so that you can print directly to PDF files from any application. There are also tools that allow you to upload documents in common file formats and convert them to the PDF format. If you have PS or EPS files, you can use the tools `ps2pdf` or `epspdf` to convert your PS and EPS files to PDF.

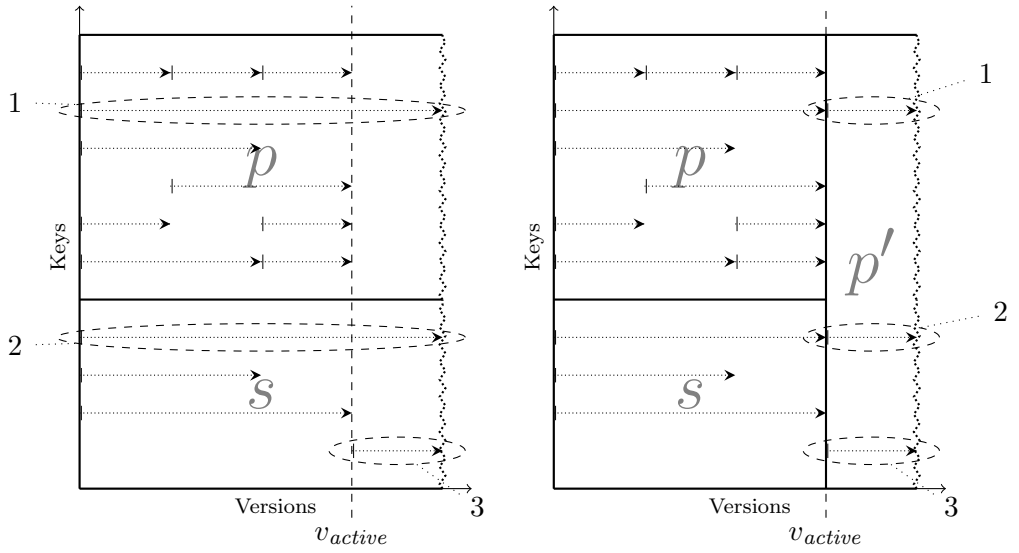
Furthermore, most newer editor programs allow you to save directly to the PDF format. For vector editing, you could try Inkscape, which is a new open source WYSIWYG vector editor that allows you to save directly to PDF. For graphs, either export/print your graphs from OpenOffice Calc/Microsoft Excel to PDF format, and then add them; or use `gnuplot`, which can create PDF files directly (at least the new versions can). The terminal type is *pdf*, so the first line of your plot file should be something like `set term pdf ...`.

To get the most professional-looking graphics, you can encode them using the TikZ package (TikZ is a frontend for the PGF graphics formatting sys-

Figure 3.2: Eeyore, or Ihaa, a very sad donkey.

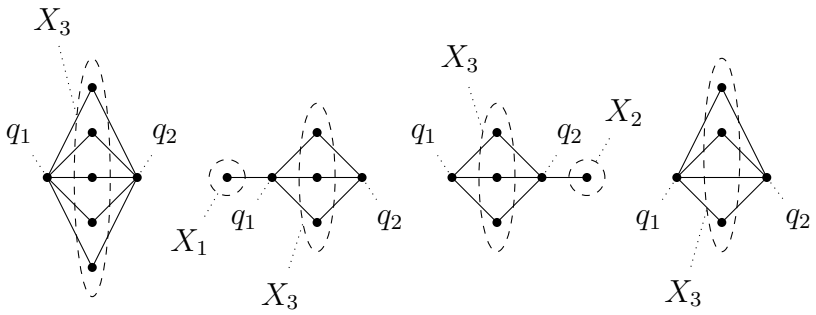
tem). You can create practically any kind of technical images with TikZ, but it has a rather steep learning curve. Locate the manual (`pgfmanual.pdf`) from your LaTeX distribution and check it out. An example of TikZ-generated graphics is shown in Figure 3.3.

Another example of graphics created with TikZ is shown in Figure 3.4. These show how graphs can be drawn and labeled. You can consult the example images and the PGF manual for more examples of what kinds figures you can draw with TikZ.

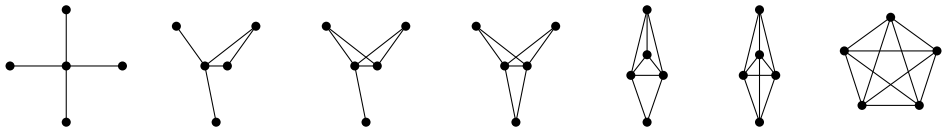


(a) Before (b) After

Figure 3.3: Example of a multiversion database page merge. This figure has been taken from the PhD thesis of Haapasalo Haapasalo [2010].



(a) Examples of obstruction graphs for the Ferry Problem



(b) Examples of star graphs

Figure 3.4: Examples of graphs draw with TikZ. These figures have been taken from a course report for the graph theory course Göös et al. [2010].

Chapter 4

Methods

You have now stated your problem, and you are ready to do something about it! *How* are you going to do that? What methods do you use? You also need to review existing literature to justify your choices, meaning that why you have chosen the method to be applied in your work.

If you have not yet done any (real) metholodogical courses (but chosen introduction courses of different areas that are listed in the methodological courses list), now is the time to do so or at least check through material of suitable methodological courses. Good methodologial courses that consentrates especially to methods are presented in Table 4.1. Remember to explain the content of the tables (as with figures). In the table, the last column gives the research area where the methods are often used. Here we used table to give an example of tables. Abbreviations and Acronyms is also a long table. The difference is that longtables can continue to next page.

Code	**Name**	**Methods**	**Area**
T-110.6130	Systems Engineering for Data Communications Software	Computer simulations, mathematical modeling, experimental research, data analysis, and network service business research methods, (agile method)	T-110
Mat-2.3170 Simulation (here is an example of multicolumn for tables)		Details of how to build simulations	T-110
S-38.3184	Network Traffic Measurements and Analysis	How to measure and analyse network traffic	T-110

Table 4.1: Research methodology courses

Chapter 5

Implementation

You have now explained how you are going to tackle your problem. Go do that now! Come back when the problem is solved!

Now, how did you solve the problem? Explain how you implemented your solution, be it a software component, a custom-made FPGA, a fried jelly bean, or whatever. Describe the problems you encountered with your implementation work.

Chapter 6

Evaluation

You have done your work, but that's[1] not enough.

You also need to evaluate how well your implementation works. The nature of the evaluation depends on your problem, your method, and your implementation that are all described in the thesis before this chapter. If you have created a program for exact-text matching, then you measure how long it takes for your implementation to search for different patterns, and compare it against the implementation that was used before. If you have designed a process for managing software projects, you perhaps interview people working with a waterfall-style management process, have them adapt your management process, and interview them again after they have worked with your process for some time. See what's changed.

The important thing is that you can evaluate your success somehow. Remember that you do not have to succeed in making something spectacular; a total implementation failure may still give grounds for a very good master's thesis—if you can analyze what went wrong and what should have been done.

[1] By the way, do *not* use shorthands like this in your text! It is not professional! Always write out all the words: "that is".

Chapter 7

Discussion

At this point, you will have some insightful thoughts on your implementation and you may have ideas on what could be done in the future. This chapter is a good place to discuss your thesis as a whole and to show your professor that you have really understood some non-trivial aspects of the methods you used...

Chapter 8

Conclusions

Time to wrap it up! Write down the most important findings from your work. Like the introduction, this chapter is not very long. Two to four pages might be a good limit.

Bibliography

Aalto University Library. Making a bibliography. webpage, March 2 2011a. http://otalib.aalto.fi/en/instructions/guides/citations/bibliography/. Accessed 4.2.2011.

Aalto University Library. webpage, March 2 2011b. How to find - brief guide to finding publications and information. Accessed 4.3.2011.

B. Becker, S. Gschwind, T. Ohler, Bernhard Seeger, and Peter Widmayer. An asymptotically optimal multiversion B-tree. *The VLDB Journal*, 5(4): 264–275, 1996.

Mika Göös, Tuukka Haapasalo, and Leo Moisio. Finding hard instances of the ferry problem. Course report for the course T-79.5203/S-72.2420 Graph theory, Aalto SCI, 2010.

Tuukka Haapasalo. *Accessing Multiversion Data in Database Transactions*. PhD thesis, Department of Computer Science and Engineering, Aalto University School of Science and Technology, Espoo, Finland, 2010. http://lib.tkk.fi/Diss/2010/isbn9789526033600/.

Tuukka Haapasalo, Ibrahim Jaluta, Bernhard Seeger, Seppo Sippu, and Eljas Soisalon-Soininen. Transactions on the multiversion B^+-tree. In *Proceedings of the 12th Conference in Extending Database Technology*, pages 1064–1075, 2009.

Taavi Hupponen, Kimmo Karlsson, Jani Laitinen, Oskar Ojala, Antti Pirinen, Esa Seuranen, Laura Takkinen, Boris von Loesch, and Tor Arne Vestbø. TeXlipse plugin for Eclipse, 2010. http://texlipse.sourceforge.net/. Accessed 25.2.2011.

The Eclipse Foundation. Eclipse, 2011. http://eclipse.org/. Accessed 25.2.2011.

Appendix A

First appendix

This is the first appendix. You could put some test images or verbose data in an appendix, if there is too much data to fit in the actual text nicely.

For now, the Aalto logo variants are shown in Figure A.1.

(a) In English

(b) Suomeksi

(c) På svenska

Figure A.1: Aalto logo variants