

Computing Science and Mathematics
University of Stirling

Smart contracts on Hyperledger Fabric

Yordan Gospodinov

Submitted in partial fulfilment of the requirements for the degree of
Computer Science

April 2019

Abstract

Summarise the dissertation within one page. Introductory headings like this are entered using the *intro* paragraph style. It is suggested that the abstract be structured as follows:

Problem: what you tackled, and why this needed a solution

Objectives: what you set out to achieve, and how this addressed the problem

Methodology: how you went about solving the problem

Achievements: what you managed to achieve, and how far it meets your objectives.

Attestation

I understand the nature of plagiarism, and am aware of the University's policy on this. I certify that this dissertation reports original work by me during my University project except for the following (adjust according to the circumstances):

- The technology review in section 3.2 was largely adapted from [1] plus *www.software-review.org/article9815.html*.
- The code discussed in section 3.1 was created by Acme Corporation (*www.acme-corp.com/JavaExpert*) and was used in accordance with the licence supplied.
- The code discussed in section 3.1.1 was written by my supervisor.
- The code discussed in section 3.1.1 was developed by me during a vacation placement with the collaborating company. In addition, this used ideas I had already developed in my own time.

Signature:

Date:

Acknowledgements

Acknowledge anyone who has helped you in your work such as your supervisor, technical support staff, fellow students or external organisations. Acknowledge the source of any work that is not your own.

Contents

Abstract	i
Attestation	ii
Acknowledgements	iii
1 Introduction	1
1.1 Background and Context	1
1.2 Scope and Objectives	1
1.3 Achievements	1
1.4 Overview of Dissertation	1
2 State-of-The-Art	2
3 Technical Chapters	3
3.1 First Section	3
3.1.1 First Subsection	3
3.1.2 Second Subsection	3
3.2 Second Section	4
4 Conclusion	5
4.1 Evaluation	5
4.2 Future Work	5
A User Guide	7

List of Figures

3.1 Highly Technical Diagram 4

Chapter 1

Introduction

Chapters are entered using the `\chapter` command; they start on a new page. Ordinary text is presented in 11 point Times, single-spaced, single-sided pages. In general, use the default spacing that headings and paragraphs give you. If you need to use quotes, preferably use single curly quotes ‘...’.

Italics text uses the `\textit` command, while **bold text** uses the `\textbf` command. Chapters, sections, etc. are identified by the `\label` command, and are referenced by the `\ref` command (e.g. chapter 4).

1.1 Background and Context

Give the background to your project and context of what you have done. Sections are entered using the `\section` command.

1.2 Scope and Objectives

Define the scope and objectives of your project.

1.3 Achievements

Summarise what you have achieved.

1.4 Overview of Dissertation

Briefly overview the contents of what follows in the dissertation.

Chapter 2

State-of-The-Art

Summarise current knowledge and what others have done in the various topics of your dissertation. Write for someone familiar with computing, but not necessarily expert in the particular topics of your project.

It is important to write a *critical* literature review that identifies gaps in current solutions and that clearly shows how the project was driven to address these gaps. This chapter should therefore feed into well-defined requirements for the project. Avoid a banal description of related work that does not carefully analyse its strengths and weaknesses.

Give references to other work by using citations like [4]. Use the `\cite` command to cite references. Books [1], standards [2], reports [3], journal articles [6], conference papers [4], and web pages [5] are conventionally presented in slightly different ways. If a web page does not have a date, you should give the date on which you consulted it.

Citations are created with a *thebibliography* environment and `\bibitem` commands in a ‘.bbl’ file. Unless you are willing to invest time in creating Bib_TE_X bibliographies, you can do this by hand.

Chapter 3

Technical Chapters

The body of the dissertation consists of a number of chapters named appropriately (*not* ‘Technical Chapter’). Follow a logical progression in how you present your work. This might be a time sequence of development activities, the phases of the software development cycle, the modules of your system, etc.

Appropriate chapters might be called Requirements, Design, Implementation and Evaluation. The emphasis should be on requirements, design and evaluation, with implementation details being of lesser importance. The requirements should be clearly stated, following from the client needs and weaknesses identified in the state-of-the-art review. The design should include discussion of the choices that were available and why particular decisions were made. The evaluation should relate back to the requirements, and demonstrate the extent to which these were met. Low-level material should appear in appendixes.

3.1 First Section

Subdivide your text into sections with the `\section` command.

3.1.1 First Subsection

If necessary, also use subsections. Subsections are entered using the `\subsection` command.

First Subsubsection

If you really need subsubsections, enter these using the `\subsubsection` command.

Second Subsubsection

And yet more subsubsections if need be.

3.1.2 Second Subsection

And, as required, more subsections.

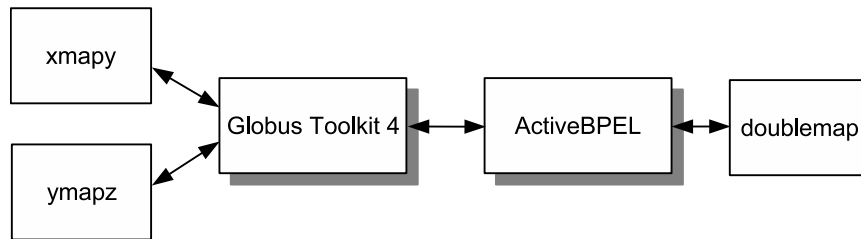


Figure 3.1: Highly Technical Diagram

3.2 Second Section

Figures are created with the *figure* environment, while tables are created with the *table* environment. They are identified by the `\label` command, and are referenced by the `\ref` command. Graphics are inserted with the `\graphic` command. Captions are entered using the `\caption` command. As an example of a figure, consider figure 3.1.

The native format for \LaTeX graphics is EPS (Encapsulated PostScript). Graphical editors are usually capable of producing EPS. When outputting to PDF (Portable Document Format), the native graphics format is also PDF. Conversion of EPS to PDF is supported by a number of TeX toolsets.

Chapter 4

Conclusion

4.1 Evaluation

If you do not have a separate chapter on testing, explain here in detail how you went about systematically testing your system. If appropriate, also include end users in your testing. Summarise your main results, and explain how you have advanced the state-of-the-art. Stand back and evaluate what you have achieved and how well you have met the objectives. Evaluate your achievements against the objectives stated in section 1.2. Demonstrate that you have tackled the project in a professional manner.

4.2 Future Work

Explain any limitations in your results and how things might be improved. Discuss how your work might be developed further. Reflect on your results in isolation and in relation to what others have achieved in the same field. This self-analysis is particularly important. You should give a critical evaluation of what went well, and what might be improved.

References

- [1] D. Greene and P. C. Williams. *Linear Accelerators for Radiation Therapy*. IOP Publishing Ltd., Bristol and Philadelphia, 1997.
- [2] ISO/IEC. *Information Processing Systems – Open Systems Interconnection – LOTOS – A Formal Description Technique based on the Temporal Ordering of Observational Behaviour*. ISO/IEC 8807. International Organization for Standardization, Geneva, Switzerland, 1989.
- [3] J. Jacobson and O. Andersen. Software controlled medical devices. Technical Report SP-Rapport 1997:11, European Network of Clubs for Reliability and Safety of Software, Apr. 1997. ISBN 91-7848-669-6.
- [4] Ji He and K. J. Turner. Specification and verification of synchronous hardware using LOTOS. In J. Wu, S. T. Chanson, and Q. Gao, editors, *Proc. Formal Methods for Protocol Engineering and Distributed Systems (FORTE XII/PSTV XIX)*, pages 295–312, London, UK, Oct. 1999. Kluwer Academic Publishers.
- [5] K. J. Turner. World-wide Environment for Learning LOTOS. <http://www.cs.stir.ac.uk/well/>, June 2000.
- [6] K. J. Turner. Representing and analysing composed web services using CRESS. *Network and Computer Applications*, 30(2):541–562, Apr. 2007.

Appendix A

User Guide

The appendixes should contain reference material or detailed material that would detract from the flow in the body of the dissertation. Appendixes might include a user guide, a list of abbreviations, detailed program descriptions, etc. Appendixes are introduced with the `\appendix` command. Appendix headings otherwise use `\chapter`, `\section`, etc. as usual.