Complexity measures for system architectures - an outline for thesis

Matti J Kinnunen

July 27, 2005

Abstract

This thesis is about measuring complexity of system architectures. We first take a detailed look at different ways of representing system architectures. Then, we take a detailed look at different ways of measuring complexity. Next, we synthesize a suitable way of measuring complexity of system architectures. In order to check the practicality of our measurement methods, we implement them in the Crawley-machine and apply the measures to several different already modeled system archictures. In order to check the validity of measurements, we also ask a group of experts to estimate the complexities of the architectures. Finally, we conclude the thesis.

1 Structure of the thesis

This section presents the outline of the thesis. We briefly discuss the preliminary contents of each section of the thesis, list the already known sources of information, and also give a rough estimate of the work amounts needed[1].

1.1 Chapter 1: Introduction

In introduction we introduce the problem. Why is measuring the complexity of system architectures important? Has it been measured before? How? Is measurement hard and if yes, why? We also introduce some preliminary examples of system architecture, which we will use as simple case studies throughout the thesis.

We also explain the structure of the thesis.

[1] the work amount estimates are relative, not absolute. Estimated amount of pages is 77, days is 67. TIme available (24 units, 4.5 months, 8 hours per day) is 78 days. Should be doable.

Legth 5 pages

Work amount 3 days

References None at the moment

1.2 Chapter 2: What is a system architecture

This section defines what we mean by system architecture. What belongs to system architecture description and what not? How to define the system boundaries? Do we have to make distinctions between software and hardware when defining system architectures. This discussion is necessary for discussing the different ways of representing the system architectures later.

Legth 5 pages

Work amount 3 days

References Rechtin, Dori, Crawley, i-Logix material, ThreeSL-material, Eppinger, Gerrit Müller, etc

1.3 Chapter 3: System architecture representation methods

This section presents different ways of representing system architectures. We want that our methods of measuring the complexity are applicable to many different representation methods.

1.4 Chapter 3.1: UML 2.0 and SysML

Short introduction to UML 2.0 and SysML, which are the newest members of the UML-family of languages. They are specifically meant for representing system level characteristics.

Legth 4 pages

Work amount 2 days

References Material available from http://www.uml.org/

1.5 Chapter 3.2: Cradle

Short introduction to the representation method used in the Cradle-toolset by a British company 3SL. This is important, since a) Cradle is one of the most common toolsets in the industry and the main competitor of UML, and b) NASA has selected Cradle as their "standard requirements management and systems engineering tool for Constellation - mission to Mars" (at least this is what 3SL claims).

Legth 4 pages

Work amount 2 days

References Material available from http://threesl.co.uk/

1.6 Chapter 3.3: Object process methodology

Short introduction to OPM as defined by Dori. Necessary as this is one of the clearest ways of representing system architectures, and also the conceptual basis for the next framework.

Legth 5 pages

Work amount 3 days

References Book by Dori, lecture material by Crawley

1.7 Chapter 3.4: Crawley-machine (Object process Network)

Introduction to the design and implementation of the Crawley-machine. This is necessary, since we intend to use the Crawley-machine as our experimentation platform, when implementing the complexity measures.

Legth 5 pages

Work amount 3 days

References Ben Koo's thesis

1.8 Chapter 4: Theories of complexity

In this section we discuss the existing ways of measuring complexity

1.9 Chapter 4.1: Kolmogorov complexity

Kolmogorov complexity is the main result in the field of algorithmic information theory. In this section we will describe the main ideas and application of Kolmogorov complexity. We will also discuss how the theory can (or cannot) be applied to measuring the complexity of system architecture descriptions.

Legth 6 pages

Work amount 5 days

References Li: Introduction to Kolmogorov complexity and its applications; also some sources from http://www.idsia.ch/ marcus/kolmo.htm and various articles.

1.10 Chapter 4.2: Decriptive complexity

Descriptive complexity is a special formulation of Kolmogorov complexity. It measures the complexity of a problem in terms of the richness of logical languages needed to describe the problem. In this section, we will explain the basic ideas of descriptive complexity, and see how it can (or cannot) be applied in measuring the complexity of system architectures.

Legth 6 pages

Work amount 5 days

References Immerman: Decriptive complexity ; also some articles

1.11 Chapter 4.1: Function point approach

In this section, we will introduce the function point approach, which is commonly used in describing the complexity of software systems. The idea is to assing function points to several constructs and operations of the software. The more points a piece of software gets, the more complex it is said to be. We will also see how it can (or cannot) be applied in measuring the complexity of system architectures.

Legth 4 pages

Work amount 4 days

References Not known yet, but should be easy to find - this is a well documented topic

1.12 Chapter 4.1: Graph approach

At least OPM and Crawley-machine store the system architecture descriptions as graphs. In this section we will describe ways of measuring the complexity of such graphs. In mathematics, the graph theory is well developed - we will just find out the main results and apply them.

Legth 4 pages

Work amount 4 days

References Not known yet, but should be easy to find - this is a well documented topic

1.13 Chapter 5: System architecture complexity measures

In this section, we will synthesize our own way of measuring the complexity of system architecture descriptions. This synthesis is based on the research done in the previous chapters.

Legth 8 pages

Work amount 15 days

References None, original work

1.14 Chapter 6: Case studies

In this section, we apply our complexity measures to some already modeled system architectures. We will also verify the results by asking a group of experts to estimate the complexity of the architectures.

Legth 20 pages

Work amount 15 days

References None, original work except references to the exisiting models

1.15 Chapter 7: Conclusion

Concluding notes and description of further research.

Legth 5pages

Work amount 5 days

References None, original work