공학석사학위논문

Entropy-based Metric for System Architecture Robustness and Expert Consistency: Formulation and Applications

시스템 아키텍쳐 강건성 및 현업 엔지니어 일관성 평가를 위한 엔트로피 기반 측정법: 공식화 및 응용

2019 년 2 월

서울대학교 대학원

산업공학과

최 민 규

Entropy-based Metric for System Architecture Robustness and Expert Consistency: Formulation and Applications

지도교수 서 은 석

이 논문을 공학석사 학위논문으로 제출함

2019 년 2 월

서울대학교 대학원

산업공학과

최 민 규

최민규의 공학석사 학위논문을 인준함

2019 년 2 월

위 원 장 ______ 홍 유 석 ______ (인)

부위원장 ______ 서 은 석 ______ (인)

위 원 ______ 문 일 경 ______ (인)

Abstract

Entropy-based Metric for System Architecture Robustness and Expert Consistency: Formulation and Applications

Minkyu Choi
Department of Industrial Engineering
The Graduate School
Seoul National University

This thesis proposes an entropy-based metric which quantifies complex system architecture robustness to different decomposition perspectives for resolving varying stakeholders' architectural preferences during the critical stages of the system architecting process. The newly developed metric aims to identify architectures that are robust to different decomposition perspectives by quantifying pairwise comparisons between two different architectural decompositions that may arise from the system architecting process. While system architects typically rely on decomposing a system into its constituent functions and subfunctions, the architecture of a complex system may be interpreted differently by various stakeholders throughout the value chain, which can result in several different system decomposition perspectives, including, but not limited to, assembly or maintenance-based decomposition preferences. As such, the various modular configurations should be quantitatively assessed for the development of an architecture that is robust for different perspectives. The newly proposed module diffusion index adapts entropy, a statistical mechanics concept, to quantify the level of re-arrangement that is required for a module's components to be reassigned to another decomposition perspective as a means of assessing an architecture's robustness to different stakeholder requirements. Two feasibility studies were conducted to observe how the newly proposed metric evaluates decomposition perspectives of three different architectures to find a perspective-robust architecture, and to assess the consistency at which industry professionals decompose a given architecture to different perspectives. The proposed metric aims to assist system architects as a quantitative evaluation criterion for analyzing different system architecture concepts during the early engineering phases of complex system design.

Keywords: Systems thinking, System architecture, Modularity
Student Number: 2017-29435

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Complex systems with long lifecycles need to accommodate for numerous, often conflicting requirements from different stakeholders involved. System architecting is one of several critical phases in the development of such complex systems where the functional requirements are allocated to cyber-physical elements that constitute an overall system concept. However, today's engineering systems are being developed with increasing complexity in order to meet the stringent and evolving demands of numerous stakeholders throughout the value chain (Sinha & de Weck, 2016).

The architecture of such a complex system cannot be fully articulated under a single perspective; rather, the architecture can be viewed under different perspectives depending on the relevant stakeholders throughout different time frames in the system lifecycle. All of these system lifecycle considerations must be considered during the system architecting phase. To this end, it is in every stakeholder's best interests to design a system that is robust to different stakeholder perspectives.

The desktop computer is an example of a system that is robust to different decomposition perspectives (Sinha & Suh, 2018). Its mainframe contains a bus-modular motherboard that can house various components such as the CPU, storage, and memory, in addition to input/output devices such as the monitor, printer, keyboard and mouse, executing their respective functions that constitute the entire desktop, which can be installed, uninstalled, and serviced with ease. However, most complex systems cannot be decomposed as straightforwardly as the desktop computer. For a complex multibillion-dollar system such as an offshore platform, different design teams are each tasked with engineering the structure, piping, electrical and instrumentation (E&I), heat, ventilation and air conditioning (HVAC), and mechanical equipment. Manufacturing teams divide the platform into separate modules that are constructed in different areas of the shipyard before they are integrated onto the main deck. However, piping, E&I, and HVAC systems are highly distributed throughout the entire plant and may require coordination among multiple manufacturing teams and their corresponding subcontractors, causing organizational challenges in channeling the various interfaces between engineering and manufacturing organizations during the system integration phase. On the other hand, those tasked with operating and maintaining the platform focus on accessibility of mission critical components for ease of maintenance, thus adding an extra dimension in designing and manufacturing a system which must also be maintainable throughout the system's lifecycle. It is therefore in the system architect's best interests to take into consideration various stakeholders' perspectives, whether it be from functional, assembly, or maintenance-based entities, to develop an architecture that can be acceptable to these different perspectives.

Robustness in engineering design has traditionally been defined as the ability of a system to maintain performance within its specified design parameters under a wide range of uncertainties and perturbations. To this end, robustness of architecture to different stakeholders' decomposition perspectives is the ability of the architecture to facilitate the various stakeholders' requirements with minimal restructuring of its consistent modules. This thesis proposes an entropy-based architecture assessment metric – the module diffusion index (MDI) – that is able to quantify pairwise comparisons between two decomposition perspectives of a given system architecture. Since stakeholders have differing, often conflicting requirements and viewpoints for a given complex system, their resulting system decomposition perspectives, whether it be from a function, assembly viewpoint or otherwise, would vary widely in the nature at which they are decomposed. This metric measures the robustness of the system architecture with respect to various stakeholder decomposition perspectives. Utilizing this metric during the system decomposition effort and concept selection stage would allow system architects to select an architecture that can satisfy the requirements of various stakeholders.

The proposed metric was subsequently utilized in two feasibility case studies that adapt the MDI to quantify robustness of architecture for system decomposition perspectives, and to measure the consistency at which industry experts decompose a given architecture's decomposition perspectives. From the first case study, the MDI identified a robust architecture to various decomposition perspectives, and the second study adapted the MDI to identify a consistently decomposable perspective of a given architecture in an empirical, participant-based decomposition exercise.

The rest of this thesis is organized as follows: Chapter 2 discusses prior literature related to system architecture generation and assessment; Chapter 3 introduces the module diffusion index; Chapter 4 discusses a case study using three different clock architectures that analyzes the robustness of the architecture using the newly developed MDI metric; Chapter 5 discusses another case study that involves expert evaluation of system decomposition where MDI is utilized as a consistency quantification metric; finally, conclusions and future works are discussed in Chapter 6.

Chapter 2

Literature Review

2.1 System Architecture Development and Selection

System architecture represents the mapping between a set of functional requirements of the system to the physical or cyber-physical elements as well as the intricate and complex interconnections between them, amalgamating into a definitive form (Crawley, Cameron, & Selva, 2015). Since several architectures can be derived from an identical set of solution-neutral functional requirements, system architecting does not yield a unique solution; rather, it is a highly iterative process of concept generation and refinement, which eventually terminates with the selection of a finalized system architecture (Kossiakoff & Sweet, 2005b; Rechtin & Maier, 2010). Architecting phases of a complex system is a critical step in the development process due to the high cost and long lifecycle of the system, since any downstream engineering changes can result in costly change propagations that may affect large portions of the system (Giffin et al., 2009).

A system architecture can be visualized and manipulated in a number of different formats. The design structure matrix (DSM), an n x n matrix that maps the different interconnections among product elements, tasks, processes, and organizations, is a commonly used tool for analyzing and visualizing system architectures (Eppinger & Browning, 2012). The DSM can also be manipulated using a wide variety of modularization and partitioning algorithms to aid in the system decomposition process (Borjesson & Hölttä-Otto, 2012; Yu, Yassine, & Goldberg, 2007). Functions of a system architecture can also be presented using the function structure method, which organizes an architecture into a hierarchical set of functions and subfunctions which can be expanded into a block diagram with associated inputs and outputs (Ulrich, 2003). The mapping of a functional requirement to a physical form can be executed using qualify function deployment (QFD), and the resulting deliverable can be analyzed for conflicting requirements, engineering characteristics, and missing or overlapping interconnections (Revelle, Moran, & Cox, 1998).

Controlled convergence approach is employed in the generation and evaluation of system architectures to thoroughly and exhaustively explore the design space (Pugh, 1991). Various approaches are at the system architect's disposal to select a system architecture concept for further development. Analytic hierarchy process (AHP) can be used as a system architecture selection tool, which makes quantitative assessments based on multiple selection criteria and the associated preference levels with respect to each concept (Saaty, 1988). Careful attention must be placed in monitoring the consistency of selection criteria when evaluating each decision unit, since quantifying inherently subjective and qualitative criteria may introduce unintended inconsistencies. In past literature, AHP has been employed on architecture selection of a variety of products, including automotive components (Hambali, Sapuan, Ismail, & Nukman, 2009) and software development projects (Zhu, Aurum, Gorton, & Jeffery, 2005). Data envelopment analysis (DEA) is an optimization-based decision making tool that maximizes the efficiency of a decision making unit (DMU) based on a linear objective function and constraints (Ramanathan, 2003). Due to its linear nature, DEA can be formulated as a simplex method, which can be computed efficiently with conventional algorithms (Charnes, Cooper, & Rhodes, 1978). Since it can be modelled as an optimization problem, its parameters and variables can be manipulated with sensitivity analysis to see which design elements are critical to objective function maximization (Cooper, Seiford, & Zhu, 2004). Okudan and Tauhid (2008) discussed how DEA is ideally suited for concept selection tasks due to its multi-input, multi-output nature. Pugh concept selection compares a variety of potential architectures with respect to a number of pre-defined criteria. A standard, best-fit design referred to as the datum is chosen as the reference that acts as the baseline concept for which comparisons can be made with other potential concepts. While the Pugh concept selection may have quantitative elements in its preparation and evaluation, system architects must take into consideration the qualitative rationale behind each concept's scores (Frey et al., 2008).

2.2 Decomposition Perspectives of System Architecture

A system is considered modular if its constituent elements such as physical components, tasks, or organizational units are grouped into modules such that the intra-modular interactions are maximized while inter-modular interactions are minimized (Sanaei, Otto, Hölttä-Otto, & Luo, 2015; Sosa, Eppinger, & Rowles, 2007). Adopting a modular system architecture is an ideal solution for handling the complexity of development and manufacturing, as well as to mitigate various uncertainties and risks during the system's lifecycle (Baldwin & Clark, 2006). However, achieving modularity in system design may cause increase in size and lower performance (Hölttä, Suh, & de Weck, 2005). On the other hand, integral designs, whose constituent elements are grouped into loosely defined modules, have smaller product form factors and improved performance (Otto & Wood, 2001). To manage these tradeoffs effectively, system architects decompose the system into manageable chunks to distribute complexity to its constituent subsystems (Braha, Minai, & Bar-Yam, 2006). Complexity, if left unchecked without a comprehensive management plan, has a negative effect on both development time and overall costs throughout the system lifecycle (Sinha & Suh, 2018). While decomposition analyses may yield prospective modular configurations that a system may adopt, potentially reducing system integration effort, it must be noted that the cumulative complexity of the entire system in question remains unchanged (Sinha & de Weck, 2013).

However, especially for complex systems, differing modularity configurations may arise due to varying, even conflicting stakeholder requirements, resulting in "modularity for X," such as modularity for design, modularity for assembly, or modularity for use (Walsh, Dong, & Tumer, 2018). Ideally, every system should be architected based on a balanced set of requirements and specifications, and no stakeholder viewpoint should have complete dominance over the others (Kossiakoff & Sweet, 2005a, 2005b). However, due to the inherent complexity of today's engineering systems, its architectural viewpoints of a given system may vary depending on the stakeholder. Despite the fact that there are numerous methods of evaluating complex system architectures, the involved stakeholders hold differing perspectives as to how they view the same system in question. The Department of Defense Architecture Framework (DoDAF) discusses the various viewpoints of the architecture of defense systems, ranging anywhere from systems, operations, to project viewpoints (Chief Information Officer, 2010). Suh, Chiriac, and Hölttä-Otto (2015) further analyzed different decomposition perspectives using a printer system which were configured into function, assembly, and maintenance-based modular arrangements at two levels of granularity. Varying modularity values were observed based on the different decomposition perspectives, indicating that these perspectives exhibited different modular configurations. These different perspectives from various organizational elements are one of the causes of interface misalignment issues during handover from engineering to production, as well as engineering challenges during the development of product families which may cause unintended change propagations resulting from differing perspectives (Gu & Sosale, 1999; Sosa, Eppinger, & Rowles, 2004).

2.3 Quantitative System Architecture Assessment

Numerous metrics have been proposed to quantitatively assess system architecture attributes during concept development. Various modularity metrics have been introduced, such as the modularity index, minimum description length (MDL), and singular value modular index (SMI) (Guo & Gershenson, 2007; K. Hölttä-Otto, Chiriac, Lysy, & Suh, 2012; Yu et al., 2007). While the precise formulations may vary, modularity metrics generally assess the level of intra-cluster interactions versus inter-cluster interactions, such that the system is considered more modular when the former is more quantifiably significant than the latter (K. Hölttä-Otto et al., 2012). Various complexity metrics were developed as an effort to quantify the component, interface, and architectural complexities as a method of assessing complexity attributes in system architectures (Kim, Kwon, Suh, & Ahn, 2016; Sinha & de Weck, 2013). System complexity can be distributed throughout decomposed subsystems of similar complexity as a measure of managing complexity (Sinha & Suh, 2018). More recently, integrative complexity was introduced by Sinha, Suh, and de Weck (2018) as an alternative measure to quantify system modularity which assesses the level of integrative work needed to assemble the various modules of a system's architecture.

While system architects have a number of metrics at their disposal that measure various system attributes such as complexity and modularity, few such metrics exist in comparing one modular configuration from another for a fixed system architecture. Likeliness index introduced by Thebeau (2001) in the Idicula-Gutierrez-Thebeau Algorithm (IGTA) assessed the similarity of two modular configurations in an effort to identify the most consistent result for the clustering algorithm. While this index uniquely makes pairwise comparisons of two modular configurations, the index is a summated element-wise dot product of a module from the first DSM to another module from the second DSM. As such, it is limited in capturing how one decomposition perspective may evolve into another throughout the value chain with respect to varying levels of stakeholder involvement.

2.4 Research Gap Analysis

Despite numerous literatures from both academia and industry regarding system architecture perspectives, little has been discussed in formulating a quantitative assessment of these perspectives. Adapting such a metric would have numerous benefits during complex system development, from assessing the system architecture robustness to different perspectives, to evaluating consistency of subject matter experts' decomposition of complex systems. Furthermore, past literatures have evaluated different decomposition perspectives using a modularity metric, a quantitative assessment of a system decomposition effort that is formulated based on the notion of rewarding intra-modular interactions and penalizing inter-modular interactions. Existing modularity metrics are infeasible for comparing and evaluating between two different decomposition perspectives because similar modularity metric values can be obtained from vastly different decomposition efforts and vice versa. Therefore, using the newly developed MDI metric in conjunction with existing modularity metrics will provide a more holistic evaluation of different architecture decomposition perspectives. This thesis adapted the newly formulated MDI to compare three different architectures to investigate whether the metric can correctly identify a robust system architecture to different decomposition perspectives. An empirical study was also conducted to examine how industry practitioners decompose a given system where the MDI is adapted to analyze the accuracy and consistency of the decomposition results.

Chapter 3

Entropy-based Metric Development

3.1 Metric Development Overview and Background

This chapter discusses the formulation of how the decomposed architectures can be quantified by the entropy-based assessment. The decomposed system perspectives are paired with the baseline decomposition perspective for the entropy-based assessment. The overall layout of the process is illustrated in Figure 3.1. During the system architecting phase, a single system can be decomposed differently by many stakeholders in the value chain. Out of numerous decomposition perspectives that may arise during system architecting effort, three main perspectives will be discussed in this thesis: function, assembly, and maintenance. These perspectives are respectively preferred by the following stakeholders in the value chain.

Design teams: an engineering organization's architectural preferences best align with the function-based decomposition, where the system elements are arranged into subsystems that carry out a specific function as detailed in the functional requirements of the system. Many engineering teams are grouped into disciplines that coincide closely with respect to a system's various functions, such as structure, piping, and electrical systems (Sosa, Eppinger, & Rowles, 2003).

Manufacturing teams: to maximize efficiency in the production line, the product is analyzed thoroughly to identify which chunks can be produced independently on separate lines, and the resulting configuration represents the assembly-based decomposition. For efficient production with minimal inter-cluster interfaces, the system should be rearranged to reflect the modularity of the resulting configuration.

Maintenance engineers: the preferences of service engineers result in the maintenance-based decomposition, which is configured for ease of disassembly and efficient servicing effort. As such, components of similar lifespans and failure rates that are located adjacent to each other may be grouped together as modules to be removed according to repair schedules. Some replaceable components may also be modularized in accordance to different safety stock levels and replenishment lead times.

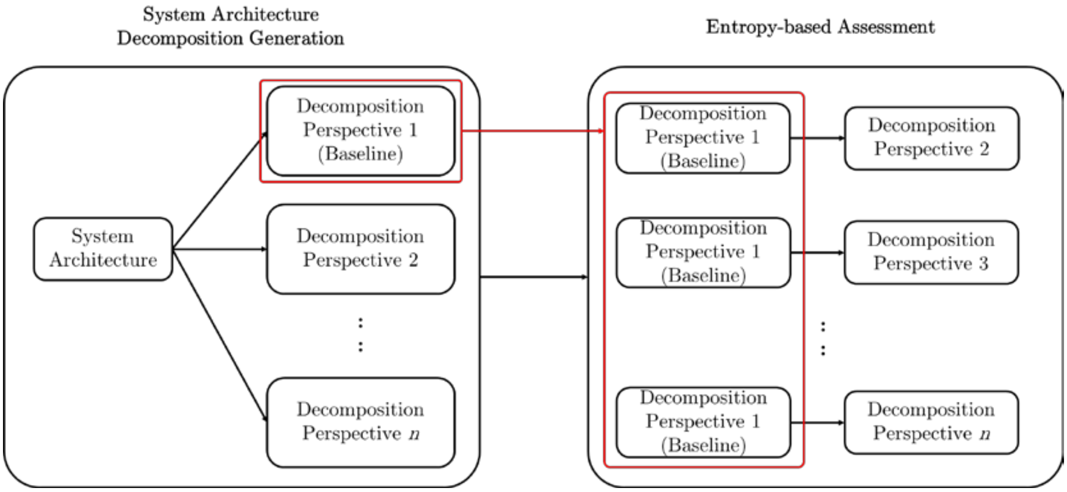


Figure 3.1: Entropy-based system architecture assessment overview

3.2 Module Diffusion Index Formulation

The main idea behind the module diffusion index (MDI) stems from entropy, a statistical mechanics concept. Entropy is a quantitative assessment of the level of disorderliness of a given system, which can alternatively be interpreted as the degree at which a system occupies different states (Schwabl, 2006). Ahn, Choi, and Suh (2018) adapted this idea as the module diffusion index (MDI) to calculate the amount of diffusion that occurs if one modular configuration is arranged into another. Entropy can be expressed with (3.1), which describes the amount of information that is required to transition from some original state to a new state k, where k_b is the Boltzmann constant and p_k is the probability of the occurrence of state k during system fluctuation such that $\sum_k p_k = 1$, and $\mathbb{E}_k[\bullet]$ is the expectation operator. Since p_k refers to the probability of state transition to some finitely possible state k, the expectation operator is replaced as follows: $\mathbb{E}_k[\ln(p_k)] = \sum_k p_k \ln(p_k)$. The quantification of state transition from the definition of entropy was adapted in this thesis to quantify the difference between two modular arrangements representing two stakeholder perspectives.

$$S = -k_B \mathbb{E}_k[\ln(p_k)] = -k_B \sum_k p_k \ln(p_k) \tag{3.1}$$

For convenience and metric scaling purposes, the Boltzmann constant is normalized to $k_b = 1$. Thus, the amount of diffusion of module i from decomposition perspective A to module k of decomposition perspective B is formulated as shown in (3.2), where $p_{i,k}$ is the probability of a component from module i from perspective A diffusing to module k of perspective B, and T_i is the total number of possible modules of perspective B onto which a component from module i of perspective A can diffuse. This equation is more intuitively written as the right-hand side of (3.2), where p_k refers to the probability of a component from module i of perspective A diffusing to some module k of perspective B, and n_i is the number of components in module i of perspective A.

$$S_i^{AB} = \sum_{k \in T_i} \left(-p_{i,k} * \ln\left(p_{i,k} \right) \right) = -\sum_{k=1}^{n_i} p_k \ln p_k \qquad (3.2)$$

Since p_k is defined as the probability of some component in module i of perspective A that is found in module k of perspective B, S_i^{AB} is summed for all $k = 1$ to n_i, which is the number of modules into which components of module i diffuse. Furthermore, (3.3) further substitutes probability p_k with $1/n_i$, which equivalent to the probability that a component of module i of perspective A is diffused into some module k of perspective B that contains a component from perspective A module i.

$$S_i^{AB} = -\sum_{k=1}^{n_i} (1/n_i) \ln(1/n_i) \qquad (3.3)$$

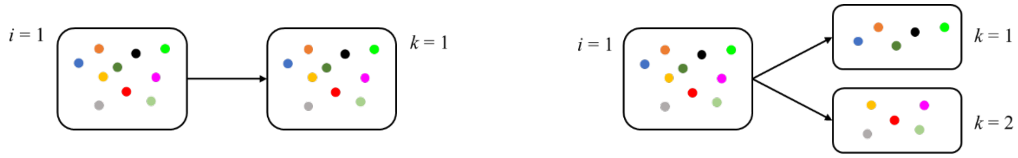
The adaptation of entropy into the new metric is now complete. However, the module diffusion index must be scaled in such a way that it is intuitively comprehensible and must also describe the diffusion of the entire modular configuration. To this end, the MDI for module i of perspective A is defined as (3.4), where the exponent scales the minimum value of MDI to 1.

$$\mathrm{MDI}_i^{AB} = e^{S_i^{AB}} \qquad (3.4)$$

The total MDI for the entire system decomposition perspective A is calculated by summation of all MDI_i^{AB} values for all i modules from perspective A, where m is equal to the number of modules in perspective A. The weight w_i is calculated by e_i / N, where e_i is the number of components assigned to module i of perspective A and N is equal to the total number of components in the system.

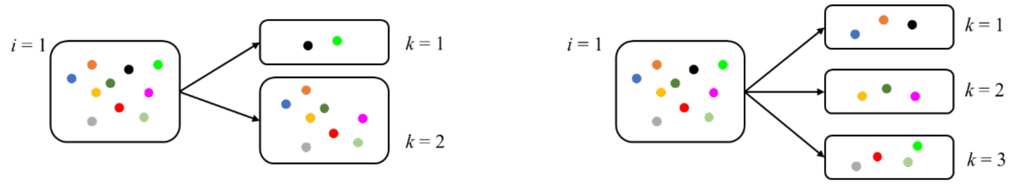
$$\mathrm{MDI}_{total}^{AB} = \sum_{i=1}^{m} \left(w_i \cdot \mathrm{MDI}_i^{AB} \right) \qquad (3.5)$$

A calculated $\mathrm{MDI}_{total}^{AB}$ value is a real number valued between 1 and number of modules in perspective B, $n_{\max}^{B}$.



(a) $S_1^{AB} = -1 * \ln(1) = 0.000$
$\mathrm{MDI}_1^{AB} = e^{0.000} = 1.000$

(b) $S_1^{AB} = -((\frac{1}{2}) * \ln(\frac{1}{2}) + (\frac{1}{2}) * \ln(\frac{1}{2})) = 0.693$
$\mathrm{MDI}_1^{AB} = e^{0.693} = 2$

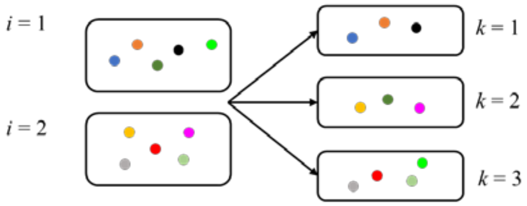


(c) $S_1^{AB} = -((\frac{1}{5}) * \ln(\frac{1}{5}) + (\frac{4}{5}) * \ln(\frac{4}{5})) = 0.500$
$\mathrm{MDI}_1^{AB} = e^{0.500} = 1.649$

(d) $S_1^{AB} = -((\frac{3}{10}) * \ln(\frac{3}{10}) + (\frac{3}{10}) * \ln(\frac{3}{10}) + (\frac{2}{5}) * \ln(\frac{2}{5})) = 1.089$
$\mathrm{MDI}_1^{AB} = e^{1.089} = 2.971$

Figure 3.2: Examples of MDI calculation

The module diffusion index between two identical decompositions, as shown in (a) of Figure 3.2 is equal to 1, since all components in the module $i=1$ of perspective A diffuse to module $k=1$. In (b), module diffusion occurs in such a way that exactly half of the components diffuse to two different modules with same number of components. In this case, $\mathrm{MDI}_1^{AB} = e^{0.693} = 2$, signifying that module $i=1$ diffuses to two different modules. Furthermore, the MDI can also capture the instance of a single module diffusing into two modules of different sizes as illustrated in (c). Here, the value of n_i is different for $k=1$ and $k=2$, since different number of components are distributed onto the two modules of perspective B, resulting in $\mathrm{MDI}_1^{AB} = e^{0.500} = 1.649$. The value is less than the MDI calculated in (b), which can be intuitively interpreted that less diffusion occurs in (c) because the components in perspective B are distributed more densely towards module $k=2$, resulting in a lower MDI_1^{AB} value. This can be further generalized as (d) for three (or more) modules, where the number of maximum modules can theoretically be equal to the number of components.



$$\mathrm{S}_1^{AB} = -\left(\left(\frac{3}{5}\right) * \ln\left(\frac{3}{5}\right) + \left(\frac{1}{5}\right) * \ln\left(\frac{1}{5}\right) + \left(\frac{1}{5}\right) * \ln\left(\frac{1}{5}\right)\right) = 0.950$$

$$\mathrm{S}_2^{AB} = -\left(\left(\frac{3}{5}\right) * \ln\left(\frac{3}{5}\right) + \left(\frac{2}{5}\right) * \ln\left(\frac{2}{5}\right)\right) = 0.673$$

$$\mathrm{MDI}_{total}^{AB} = \frac{1}{2} * e^{0.950} + \frac{1}{2} * e^{0.673} = 2.273$$

Figure 3.3: Example of MDI calculation for more than one original module

So far, the baseline configuration contained only one single module that diffused into one or more modules. Figure 3.3 shows the diffusion of a system of ten elements grouped into two modules into a different configuration of three modules. In this case, S_i^{AB} must be calculated for $i=1, 2$. For a baseline perspective of more than one module, $\mathbf{MDI}_{total}^{AB}$ is calculated as a weighted summation where $w_i = 1/2$, since both modules of perspective A contain 5 out of 10 elements each. Furthermore, it must be noted that MDI is a non-commutative property such that $\mathbf{MDI}_{total}^{AB} \neq \mathbf{MDI}_{total}^{BA}$. In fact, if $\mathbf{MDI}_{total}^{BA}$ is calculated for the diffusion in Figure 3.3, then the new MDI is equal to $\mathbf{MDI}_{total}^{BA} = 1.569$. This makes it critical in deciding which baseline decomposition perspective, since choosing the baseline as either perspective A or B may yield entirely different results. The subsequent two sections present two case studies to assess the newly formulated metric in identifying architecture robustness to decomposition perspectives and assessing the consistency of various perspective-based decomposition solutions provided by industry professionals.

Chapter 4

Case Study: System Architecture Robustness Assessment for Different Stakeholder Perspectives

4.1 Introduction

This case study investigates the module diffusion index as a measurement of architecture robustness of three mechanical clock systems with respect to different stakeholder perspectives. The primary objective of this case study was to analyze if there were any differences in module diffusion index between function-to-assembly pairwise comparisons and function-to-maintenance pairwise comparisons. The function-based perspective was chosen as the basis for comparing both the assembly and maintenance-based perspectives because design and engineering of any system precedes the assembly and servicing activities with respect to the entire lifecycle of the system. As such, it is preferable that the function-to-assembly and function-to-maintenance perspectives to be as low as possible (close to one) to minimize organizational mismatches between engineering organizations and other teams and entities responsible for downstream tasks such as assembly and servicing, and reduce the amount of change propagations resulting from technology infusion of a specific module.

The three architectures used in this case study are mechanically operated clocks consisting of approximately 40 to 50 components. Each clock architecture was analyzed by constructing a binary, symmetric, product-architecture design structure matrix to map the physical interconnections between the components. The front views of each clock and the associated DSM, modularized for the function-based perspective, is shown in Figure 4.1. To make effective apple-to-apple decomposition comparisons, the level of granularity was fixed at the component level for the construction of the DSM. Table 4.1 lists the main functions of the clock and Table 4.2 lists the number of components assigned to each module for all clock architectures.

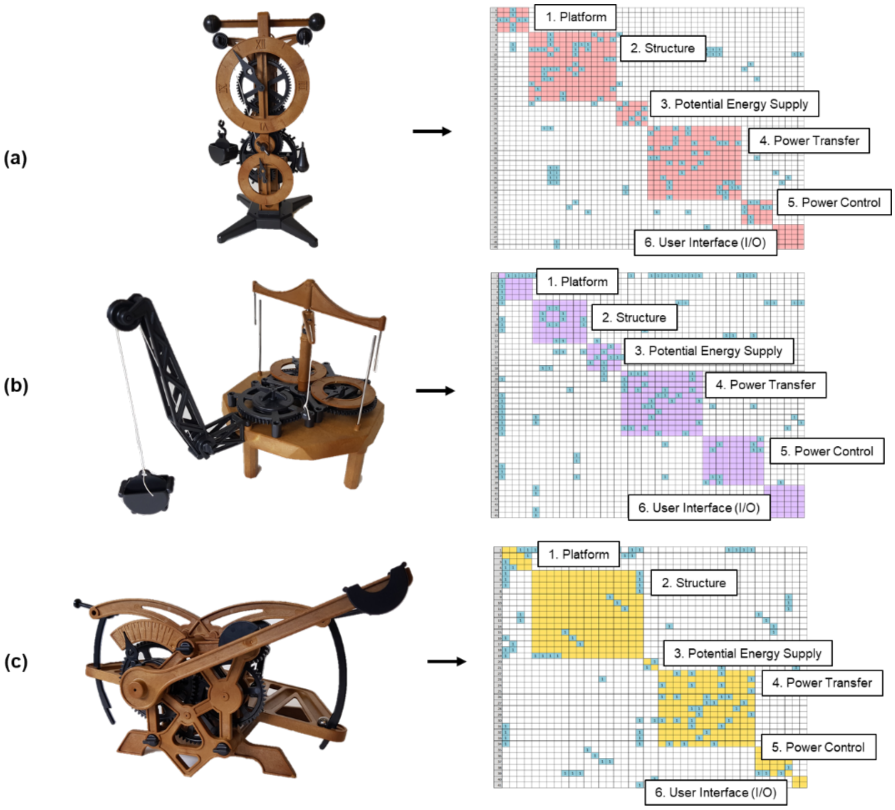


Figure 4.1: Front views and function-based decompositions for (a) VFEC, (b) FPC, and (c) CSC architectures

Table 4.1: Description of function modules

Module Name	Description of Module
1. Platform	Components provide structural stability to clock system
2. Structure	Structural components house rotating and nonstationary components together
3. Potential Energy Supply	Coin basket and associated components provide potential energy for clock operations
4. Power Transfer	Gears and shaft components transform potential energy into rotational motion
5. Power Control	Controls rotating motion provided by gears into discretized clock motion at fixed increments
6. User Interface (I/O)	Clock hands and faces display time to user

Table 4.2: Component distribution for each module and calculated weights

Module	VFEC		FPC		CSC	
	e_i	w_i	e_i	w_i	e_i	w_i
1. Platform	5	0.102	5	0.111	4	0.098
2. Structure	14	0.286	8	0.178	15	0.366
3. Potential Energy Supply	5	0.102	5	0.111	2	0.049
4. Power Transfer	15	0.306	12	0.267	13	0.317
5. Power Control	5	0.102	9	0.200	5	0.122
6. User Interface (I/O)	5	0.102	6	0.133	2	0.049
Total	49	1.000	45	1.000	41	1.000

4.2 Clock Architecture Overview

The mechanisms of the clock architectures used in the case studies are briefly summarized below.

Verge and Foliot Escapement Clock (VFEC): this clock architecture is a mechanically operated clock consisting of 49 total components. It utilizes potential energy to drive an escape wheel, which is manipulated by a pallet that controls the rotational energy into discretized clock movements. The pallet is also connected to a verge and foliot mechanism where the attached weights at each end of the foliot is used to control the clock speed by manipulating centrifugal force of the verge and foliot assembly (Roup, Bernstein, Nersesov, Haddad, & Chellaboina, 2003). This clock architecture was used for both case studies 1 and 2.

Flying Pendulum Clock (FPC): The FPC architecture is comprised of a total of 45 components which also uses potential energy to provide rotational motion of the clocks. The components of the FPC architecture are housed on a single main platform. The potential energy provides rotating motion to a steel ball which is attached to a string at the end of a crane. Two steel rods which are vertically installed at each end of the platform latch onto the steel ball which control the continuously rotating crane into discretized clock movements. As the crane and the attached ball rotate, periodic, discretized timekeeping movement for the clock is provided by the gears and shafts. (Clausen, 1883).

Congreve-style Clock (CSC): The crane translates the potential energy in the form of torque, which is used to drive the timekeeping mechanisms of the clock. To control and discretize the torque movement provided by the crane, a steel ball travels along an angled ball track which oscillates at fixed intervals. As the ball track oscillates back and forth due to the ball movement, the timekeeping hand moves in synchronization with the oscillation, resulting in clock motion (Hillis, 2000). This clock was also used in both case studies.

4.3 Stakeholders' Decomposition Perspectives

Function, assembly, and maintenance-based decomposition perspectives were analyzed in this case study so that the entire lifecycle of the system from development to operations could be captured. These three perspectives respectively reflect the preferences and requirements of the engineering offices, manufacturing teams, and maintenance engineers involved at different phases throughout the system value chain. Detailed explanations of these perspectives specific to the clock architectures are given below.

Function-based perspective: The function-based perspective decomposed the components into different functions that the clock architectures execute during operation. Typically, function-based decomposition perspectives are useful in organizing different engineering teams, each responsible for a specific discipline, such as structural or electrical engineering teams. As such, each module for this decomposition perspective can be defined by the high-level function that the module is responsible for executing, often correlating closely with a specific engineering discipline or team.

Six common functions were identified for all three clock architectures. The platform module provided the necessary structural stability and support to erect the clocks as freestanding structures. The structure module consisted of the exterior skeletal components that housed nonstationary and rotating components. The potential energy supply module was responsible for providing the clocks with the necessary energy to operate its moving components. The power transfer module consisted of gears and shafts that translated the potential energy to rotating mechanical energy to allow the clock hands to make time measures. The power control module was responsible for discretizing the continuous motion provided by the power transfer module such that the clock hands rotate in steady, incremental movements. Lastly, the input/output module was a collection of clock hands and clock faces that displays the time information to the user. Function-based decomposition was executed by carefully analyzing each component to ascertain its primary function so that it could be assigned to each module.

Assembly-based perspective: The assembly-based perspective was decomposed by adhering strictly to the build order of each architecture. Groups of components with minimal interfaces that could be independently constructed were amalgamated as different modules. Such a practice is commonly employed in the shipbuilding and offshore construction industry where various utility and process modules are constructed at different locations throughout the shipyard and installed onto the topsides of the vessel or offshore platform (Paik & Thayamballi, 2007). Efforts have also been made to take into consideration the modularity of the decomposition configuration, and modules were selected such that the inter-cluster interactions were minimized with the aims of reducing system integration effort during assembly.

While the functions remained identical, each clock architecture was decomposed differently from the assembly-based perspective to account for unique architectural elements the clocks. Structural components for all clocks were divided into several assembly modules where each module could be constructed separately before being placed into a central, parent structure or platform. Gears and shafts were grouped as one module due to the tight coupling and interactions between these components. Lastly, a tertiary components module housed the small pins, hooks, and instruments that are placed on the outermost structure of the clocks.

Maintenance-based perspective: The maintenance-based perspective organized components into structural or replaceable modules. Typically, components under the maintenance-based perspective are modularized with respect to preventative and corrective maintenance schedules for cost-effective and flexible servicing for system reliability and fleet readiness (Joo, 2009). Structural modules for the three clock architectures were permanent, non-replaceable modules that were established to aid in the prompt disassembly of structural components of the clocks, while other modules took into consideration of different maintenance, repair and overhaul (MRO) strategies that would be employed throughout the lifecycle of the system. For example, gears and shafts were typically modularized together as high-maintenance modules due to their need for frequent servicing and repair.

After each clock was decomposed into three different function, assembly, and maintenance-based viewpoints, the MDI was calculated between the function-to-assembly and function-to-maintenance decomposition perspective pairs for all three clock architectures. The rationale behind choosing the function perspective as the baseline perspective for calculating MDI stems from the temporal precedence of engineering deliverables in any complex system. The engineering organizations provide approved drawings for construction, where engineering deliverables such as shop drawings, 3D models are extensively used for construction (Eyres & Bruce, 2012). After the complex system is completed and delivered to the client, servicing organizations also utilize deliverables originating from engineering organizations, in this case, as-built drawings (Clayton, Johnson, Song, & Al-Qawasmi, 1998). Because engineering works both take precedence over either assembly and maintenance activities, and because system architecture is decided during the early phases of engineering activities, minimizing the diffusion and perspective differences between different stakeholders should be done with respect to the function-based decomposition perspective.

After the clocks were decomposed into three perspectives, MDI was calculated between the two decomposition pairs to assess whether the MDI metric is able to identify which clock architecture is most robust to varying decomposition perspectives, and whether there are any discernable patterns in how MDI^{FA} and MDI^{FM} values differ among the three architectures.

4.4 Case Study Results

Function, assembly, and maintenance-based perspectives were generated for all three clock architectures and were visualized with the DSM as shown in Figures 4.2 and 4.3. Tables 4.3 and 4.4 list the interim S and MDI values for each of the six function-perspective modules. The MDI for function-to-assembly and function-to-maintenance were calculated to assess which architecture was most robust to different perspectives. Examining the MDI results, the function-to-maintenance diffusion was higher than function-to-assembly diffusion. Furthermore, FPC clock architecture, whose MDI values scored the lowest for both perspective comparisons, was identified to be the most robust design to different decomposition perspectives.

Table 4.3: MDI for each function module and MDI_{tot}^{FA} for all architectures

Module	VFEC		FPC		CSC	
	S_i^{FA}	MDI_i^{FA}	S_i^{FA}	MDI_i^{FA}	S_i^{FA}	MDI_i^{FA}
1. Platform	0.000	1.000	0.000	1.000	1.040	2.828
2. Structure	0.876	2.401	1.074	2.926	1.171	3.225
3. Potential Energy Supply	0.950	2.586	0.673	1.960	0.000	1.000
4. Power Transfer	0.500	1.649	0.000	1.000	0.790	2.204
5. Power Control	0.000	1.000	0.530	1.698	1.055	2.872
6. User Interface (I/O)	0.673	1.960	0.868	2.381	0.693	2.000
MDI_{tot}^{FA}	1.859		1.773		2.651	

Table 4.4: MDI for each function module and MDI_{tot}^{FM} for all architectures

Module	VFEC		FPC		CSC	
	S_i^{FM}	MDI_i^{FM}	S_i^{FM}	MDI_i^{FM}	S_i^{FM}	MDI_i^{FM}
1. Platform	0.000	1.000	0.000	1.000	0.000	1.000
2. Structure	1.433	4.190	0.900	2.460	1.564	4.779
3. Potential Energy Supply	0.000	1.000	0.673	1.960	0.000	1.000
4. Power Transfer	1.310	3.704	1.011	2.749	0.690	1.994
5. Power Control	0.000	1.000	0.687	1.988	1.055	2.872
6. User Interface (I/O)	0.500	1.649	0.637	1.890	0.693	2.000
MDI_{tot}^{FM}	2.806		2.149		2.975	

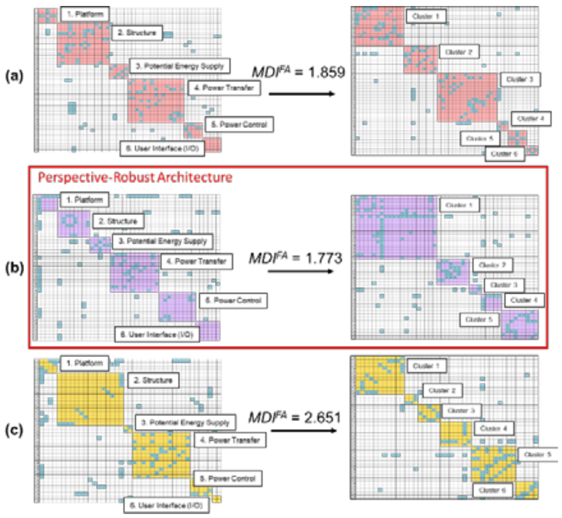


Figure 4.2: Function (left) and assembly (right) decomposition perspectives and MDI^{FA} for (a) VFEC, (b) FPC, and (c) CSC architectures

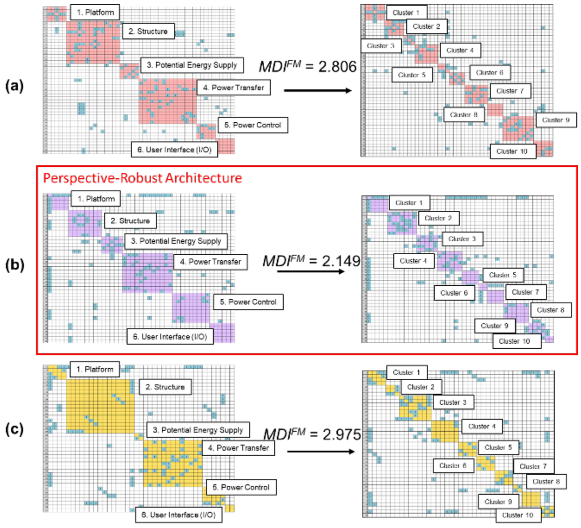


Figure 4.3: Function (left) and assembly (right) decomposition perspectives and MDI^{FM} for (a) VFEC, (b) FPC, and (c) CSC architectures

Finding 1: Function-to-maintenance diffusion was higher than that of function-to-assembly for all architectures ($\mathrm{MDI}^{FA} < \mathrm{MDI}^{FM}$). The diffusion from function-to-maintenance perspective was more significant than that of function-to-assembly perspective for all three clock architectures. This case study demonstrated that if commonly specified decomposition philosophies are employed across different architectures, the MDI is able to capture the varying levels of diffusion between the decomposition viewpoints. All of the function, assembly, and maintenance-based perspectives followed conventional decomposition practices, and the MDI was able to assess that the function-to-maintenance decomposition was more diffusive than that of the function-to-assembly decomposition.

This observation can be attributed to the specific design philosophy of the systems analyzed in this case study. These clock architectures were not designed for a long lifecycle; instead, the design focuses on allowing the users to assemble the product as easily as possible. As such, function-to-maintenance MDI (MDI^{FM}) diffuses to a greater extent than function-to-assembly MDI (MDI^{FA}) for the clock architectures, which are designed for ease of user assembly. For clock architectures, MDI captured the idea that the clocks were designed for assembly more so than they were designed for maintenance or lifecycle.

Finding 2: Clock 2 was designed with the most perspective-robust architecture, $\mathrm{MDI}_{FPC}^{FA}=\min\{\mathrm{MDI}_{VFEC}^{FA},\mathrm{MDI}_{FPC}^{FA},\mathrm{MDI}_{CSC}^{FA}\}$ and $\mathrm{MDI}_{FPC}^{FM}=\min\{\mathrm{MDI}_{VFEC}^{FM},\mathrm{MDI}_{FPC}^{FM},\mathrm{MDI}_{CSC}^{FM}\}$. The newly developed MDI was able to identify that the FPC architecture is most robust in terms of different decomposition perspectives. Close examination of the FPC architecture reveals that the function modules are highly localized into each respective location in the system, meaning that most of its functions were not distributed as piping or electrical wirings commonly found in complex systems. In fact, the clock itself was a somewhat bus-modular design, where all components necessary for clock operations were housed on top of a common platform component. This also allowed for an assembly-based perspective to closely follow the results of the function-based perspective, since each assembly module, sans a few structural key interface components, were responsible for a single function. Gears and shafts were located in a relatively accessible portion of the central platform, and the potential energy supply installed on a crane independent from the rest of the system, which also resulted in low diffusion of maintenance-based perspectives as well.

CSC architecture needed to accommodate for the house the ball rack needed for controlling the clock movement. This component was a large, distributed component that exhibited non-functional qualities such as attractive appeal. The inflated MDI values for CSC architecture properly captured the non-functional system property of the clock. Because of the sizing constraints imposed onto the system due to the large rack, a large portion the clock's critical gear assemblies and I/O module had to be compressed onto the front-facing portion of the clock, which resulted in a highly integral set of components in a localized area. This rendered each decomposition to be starkly different from one another, which resulted in noticeably large MDI when the perspectives were compared with one another.

The proposed MDI was able to identify which architecture was most robust to decomposition perspectives by comparing the various modular configurations of each architecture. Thorough examinations of each clock architecture revealed that the more robust architecture exhibited elements of bus-modularity, which resulted in a relatively continuous diffusion of function, assembly, and maintenance modules. An architecture which contained highly distributed modules resulted in high MDI values for both perspective comparison combinations.

4.5 Case Study Discussion and Summary

It was demonstrated from the case study that MDI is able to identify which module is most robust to various decomposition perspectives. This is due to the fact that the FPC architecture is housed on a bus-like central platform where sets of components such as gears and shafts are assembled in horizontal decks. This arrangement allowed close alignment between all decomposition perspectives. The CSC architecture on the other hand, featured highly non-functional design aspects to achieve a certain type of aesthetic, rendering it difficult to assign function-to-component mapping during the decomposition process. This phenomenon was exacerbated by the fact that several structural elements such as pins were required to maintain the primary structure of the architecture, resulting in a highly distributed set of elements performing the same function. This led to highly different assembly and servicing characteristics, resulting in large $\mathbf{MDI}^{FA}$ and $\mathbf{MDI}^{FM}$ values when compared to other architectures.

As stated in Chapter 3, MDI is non-commutative such that $\mathbf{MDI}^{AB}_{total} \neq \mathbf{MDI}^{BA}_{total}$, and careful consideration must be made before choosing which perspective to choose as the baseline for comparison. In this system decomposition case study, the function-based decomposition was chosen as the baseline for MDI calculation. The function-based decomposition signifies the preferences of those who are involved in engineering works, which is the most chronologically upstream task when compared to assembly and maintenance activities. Thus, the proposed metric was utilized in case study 1 to capture the temporal nature of decomposition perspective changes throughout the system value chain.

As per finding 1, MDI for function-to-maintenance was larger than for function-to-assembly. This phenomenon, spanning across all three clock architectures, can be attributed to the fact that the clock architectures themselves were not designed for maintenance-based activities over a long lifecycle. However, different results may be generated if the a highly serviceable system such as an office printer is examined, whose original equipment manufacturer (OEM) has a significant stake in maintenance and repair of its products. As such, the design intentions of the OEM may play a factor in how the MDI diffuses to different stakeholder perspectives, especially if one particular perspective is part of a firm's business portfolio.

Robustness to different decomposition perspectives were measured by quantifying the unidirectional comparison of function-to-assembly and function-to-maintenance perspectives. In other words, $\mathbf{MDI}^{FA}$ and $\mathbf{MDI}^{FM}$ assesses the robustness to system architecture perspectives between function-to-assembly and function-to-maintenance, respectively. The FPC architecture was concluded to be the most robust architecture because the architecture had the lowest $\mathbf{MDI}^{FA}$ and $\mathbf{MDI}^{FM}$ values among all three architectures considered. However, additional considerations should be addressed for assessing real-life complex systems. Such systems may need to analyze many other stakeholder perspectives, including, but not limited to, procurement teams, decommissioning teams, and end users. This may lead to system architectures with non-dominating MDI perspective comparisons. A comprehensive index consisting of a weighted summation of MDI values should be considered to address this issue of evaluating the architectural robustness to different stakeholder perspectives for complex, real-world systems.

From this case study, the newly proposed MDI was able to quantify how one decomposition perspective is different from another. The MDI quantifies the differences based on different modules' rearrangement of their components and does not take into consideration the network interconnections between the components. As such, a calculated MDI value can be interpreted as the number of modules, on average, into which a baseline module diffuses. For system architectures with intricate network and interface considerations, modularity or integrative complexity metrices can complement the MDI during the assessment of such system architecture perspectives. The adaptation of the MDI exclusively for system architecture assessment successfully identified system architectures that are most robust to different decomposition perspectives.

Chapter 5

Case Study: Expert Evaluation for Decomposition Consistency

5.1 Introduction

In a real-world engineering effort, the decomposition perspectives from the previous case study would not be prepared by a single person, but by a team of system architects and engineers. A system architect cannot be expected to possess the insight necessary to correctly decompose the various decomposition perspectives regardless of his or her engineering expertise. Since decomposition of any complex system would therefore be a collective effort, and since such a collective effort would result in some variations even within a specific decomposition perspective type, this study aims to adapt the MDI to assess how consistently a group of experts decompose various modular configurations.

Case study 2 simulates a hackathon scenario where a problem is given to a number of experts who are expected to provide novel, out-of-the-box solutions under constrained time (Briscoe, 2014). While primarily a software development effort, these short-term endeavors have been previously executed to yield new system architectural insights by Katja Hölttä-Otto et al. (2018). The goal of such an activity is to involve subject matter experts during the early phase of system development to provide insights to the system architecture that may otherwise go unacknowledged by existing system architects and engineers.

The primary objective of this study was to adapt the MDI to assess the amount of consistency for a particular decomposition perspective given that a number of engineers individually contributed in a collective effort at decomposing an architecture with respect to different perspectives. Generating a system architecture concept where engineers can reach an undisputed consensus in constructing a decomposition perspective would be valuable in assessing the feasibility and the ease of decomposition of each perspective for a given architecture.

The participants of this study were each given either a VFEC or CSC clock architecture and were assigned to decompose the given architecture into either a functional, assembly, or maintenance-based perspective. The clock architectures were first constructed and disassembled by each participant and subsequently decomposed with respect to his or her assigned perspective by utilizing the given system, assembly manual, and bill of materials (BOM). Each participant's decomposition results were recorded into an Excel file that detailed the description of each module for the architectural decomposition and the module assignment for each component. MDI was calculated between each modular configuration within the same perspective of a given clock architecture. The MDI results for function, assembly, and maintenance-based perspectives for the VFEC and CSC clocks were averaged to identify which perspective was most consistently decomposed by the participants.

The participants of this study consisted of 94 industry practitioners enrolled in the

Graduate School of Engineering Practice and the Engineering Project Management programs at Seoul National University, in addition to engineers enrolled in systems engineering employee training programs. Table 5.1 details the specific instructions given to the experts for this case study.

Table 5.1: Instructions for system decomposition perspective exercise

	System decomposition perspective exercise
Instructions	For complex system decomposition, the various module sets may be generated based on different stakeholder perspectives throughout the value chain of the system. Namely, the system may be decomposed via functional, assembly, or maintenance-based perspectives. The function-based decomposition is executed by identifying key functions that the system executes and allocates the components based on which function each component executes. The assembly-based decomposition rearranges each component of the system into tightly clustered modules to reduce system integration effort. The maintenance-based decomposition either groups the components into modules that can be easily removed, or reorganizes them into maintenance and repair schedules so that entire module can be serviced at once. The system being studied for this exercise is an educational model kit of a verge and foliot escapement clock (VFEC), or a Congreve-style clock (CSC). Each clock is a mechanically operated system that harnesses potential energy from a suspended weight and converts it to rotational energy which is used to power the clock needles for measuring time. Your task is to thoroughly study the system by building, disassembling the clock, and reading the related instructions and bill of materials (BOM) to decompose the system into your assigned decomposition perspective (functional, assembly, or maintenance). Complete the decomposition in the provided Excel spreadsheet and write detailed explanations for the considerations, thought processes and rationale behind your decomposition configuration.

The MDI of an arbitrary perspective A was calculated between all engineers involved in the decomposition of that perspective. $\mathrm{MDI}^{A}_{i_A,j_A}$ for $i_A = 1, 2, \ldots, n_A$, $j_A = 1, 2, \ldots, n_A$, and $i_A \neq j_A$, where i_A and j_A are engineers assigned to the assembly perspective and n_A is the number of engineers assigned to decompose the architecture into perspective A. This would result in a total of $n_A^2 - n_A$ possible MDI comparison pairs for the single perspective. Identical iterations are executed for all relevant perspectives. In this case study, function, assembly, and maintenance-based perspectives were analyzed for expert consistencies, and are denoted as $\mathrm{MDI}_{i_A,j_A}^{A}$, $\mathrm{MDI}_{i_F,j_F}^{F}$, and $\mathrm{MDI}_{i_M,j_M}^{M}$ respectively. Two clock architectures, VFEC and CSC, were used to examine whether there are any discernable relationships between experts' decomposition consistencies and type of decomposition for the two clock architectures.

5.2 Case Study Results

After $\mathrm{MDI}_{i_A,j_A}^{A}$, $\mathrm{MDI}_{i_F,j_F}^{F}$, $\mathrm{MDI}_{i_M,j_M}^{M}$ values were calculated for each expert pair permutation, the results were analyzed by comparing the average values of each decomposition perspective for both VFEC and CSC architectures. A smaller average MDI value for a specific perspective signifies that the experts involved were able to consistently decompose the given architecture in its stated configuration. For both clock architectures, the function-based decomposition proved to be the most consistent, followed by assembly and maintenance-based decompositions, such that $\mathrm{MDI}^{F} < \mathrm{MDI}^{A} < \mathrm{MDI}^{M}$. Table 5.2 details the experts' consistency values in the form of average MDI of the three decomposition perspectives for VFEC and CSC clock architectures.

Table 5.2: Expert consistency of different perspectives for VFEC and CSC

Clock/Perspective	Function	Assembly	Maintenance
VFEC	2.190	2.236	2.327
CSC	2.118	2.148	2.335

Function-based decomposition proved to be the most consistent when the MDI values were averaged for both architectures, while experts found it comparatively more difficult in consistently decomposing the system to either assembly or maintenance-based perspectives. This finding was confirmed for both the VFEC and CSC clock architectures. Given the generally high MDI averages for both the assembly and maintenance-based perspectives, experts appear to have exhibited dissimilar views in declaring and defining the purpose and scope of each module. Also considering that decomposition is a traditionally function-based effort, experts also appear to have had more ease in choosing the number of function-based modules and defining the purpose and scope to match the system's functional requirements. Such exercises are commonly done in the form of QFD and function structure method. This suggests that additional guidelines must be provided by the system architect to generate consistent assembly and maintenance-based decomposition perspectives. Much like the Delphi method, which is an iterative brainstorming effort, this type of system decomposition effort may require several exercises for experts' opinions to fully converge and translate into accurate decomposition configurations.

In this case study, two clock architectures were analyzed with respect to different decomposition perspectives provided by industry experts, and the consistencies of their interpretations of the architectures were assessed by using the MDI. The aim of this case study was to assess how experts perceive different architectures and different decomposition perspectives. The subjects involved in this study decomposed the function-based perspective in a more consistent manner than either assembly or maintenance-based perspectives for both architectures.

5.3 Case Study Discussion and Summary

The calculation of consistencies of experts' decomposition perspectives was clearly demonstrated by the proposed metric in this case study. MDI was able to identify that the function decomposition perspective was most consistently decomposed by experts for both VFEC and CSC architectures. Experts' decomposition solutions appeared to be more inconsistent with the assembly and maintenance-based perspectives. This coincided closely with experts' intimate knowledge general engineering practices such as QFD as well as possessing mid-level management experience in an engineering office, which are often compartmentalized into teams that focus on different engineering functions.

Should a real-world examination of decomposition perspectives yield unacceptably inconsistent results, the system architects should reassess the following areas: the information, such as the DSM, function hierarchy, or product breakdown structure that is presented to the experts; the specific instructions on how to decompose a system architecture; and finally, entire architecture to re-evaluate whether the system is inherently unable to be decomposed consistently. For the last case, the inconsistent architecture should be supplanted by another competing system architecture subject to expert assessment study.

The non-commutative property in MDI ($\mathbf{MDI}_{total}^{AB} \neq \mathbf{MDI}_{total}^{BA}$) was not utilized in this case study, since setting an appropriate baseline perspective among a group of experts is unrelated to the scope of this case study. As such, when comparing experts i and j, comparison between $\mathbf{MDI}_{i,j}$ and $\mathbf{MDI}_{j,i}$ were both conducted in the calculation of average consistencies. Conducting bi-directional pairwise comparison calculations were done for all three perspectives to ensure that every expert's decomposition solutions were used both as the baseline and as the comparison sample.

The newly developed MDI metric can be adapted to assess numerous, competing system architectures during the early phases of system development.

1. System architects initiate the assessment process by inviting a sufficient number of stakeholders from the value chain and group them into different perspective types. Typical perspective types should include, but are not limited to, function, assembly, and maintenance-based perspectives.
2. Upon providing them with the necessary information and tools with which to analyze and manipulate the system architecture, stakeholders are instructed to decompose the system with respect to their assigned decomposition perspectives. The system architects may provide the experts with the system architecture DSM, work breakdown structure, product breakdown structure, and software tools to manipulate the given information into system decomposition analyses.
3. The newly developed metric should be used to assess the differences of stakeholders among each decomposition perspective to calculate the levels of consistency for each decomposition perspective. If the calculated MDI values are too inconsistent, then the decomposition process is re-iterated for that specific

perspective. If consistency cannot be established for that perspective, necessary design changes should be made to the system architecture for the specific perspective and the process should be repeated from step 1.

4. Once consistency is established among the system perspectives, a single decomposition configuration should be chosen per perspective type.
5. After a representative set of each decomposition perspective has been established, diffusion of different perspectives should be calculated by fixing the function decomposition as the baseline for MDI calculation. Diffusion results for each architecture should be recorded for comparison.
6. This process is repeated for all the architectures under consideration for system development.
7. After the selection and assessment is completed, the MDI values for the different architectures are examined to select an architecture that registers the lowest MDI values for different decomposition perspective pairs.
8. A system architecture robust to different decomposition perspectives is chosen by using consistent expert judgment and utilizing MDI to identify perspective-robust decomposition configurations and system architectures.

This proposed entropy-based architecture assessment process can be used as a system architecture selection criterion in identifying architectures that are robust to different stakeholder perspectives and consistent in expert decompositions. The two case studies conducted were designed to encapsulate key elements of the process where the MDI should be utilized. Adopting such a process would assist in the system architect's endeavor in selecting the most appropriate architecture for to fit the functional requirements.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

In this thesis, an entropy-based module diffusion index was proposed to address the issue of assessing system architecture robustness under different viewpoints by quantifying two different modularized system architecture configurations. While many different architecture assessment approaches have been discussed in literature from quantifying system modularity to complexity, the MDI quantifies the degree at which one system architecture configuration changes to another.

Two case studies were conducted to assess the feasibility of the newly developed metric. The first case study analyzed three different mechanical clock system architectures to identify the most robust architecture for different decomposition perspectives. The MDI was able to correctly identify the most perspective-robust FPC architecture by analyzing function-to-assembly and function-to-maintenance decomposition perspective pairs. The second case study adapted the MDI as a measure for consistency among an expert-generated set of decomposition solutions for a particular perspective. The case study identified the most consistent perspective as the function-based perspective for all clock architectures, which coincides with the sample group's predominant engineering background. This metric can be used as part of an entropy-based architecture assessment process which can be used as a system architecture selection criterion.

6.2 Future Works

The MDI should be adapted in the analysis of real-world complex systems to assess how such systems' decomposition perspectives are quantified with respect to the MDI. The number of stakeholder perspectives should also be extended beyond function, assembly, and maintenance-based perspectives to encompass the entire value chain for a more comprehensive evaluation of architectural perspective robustness. The two case studies presented in this thesis used clock architectures to readily compare different architectures of similar size and scope for an apple-to-apple comparison. Applying the metric to real-world systems would not only provide further validation of the proposed metric but may also yield new patterns and correlations for further investigation.

Utilizing the MDI in development of a new modularization algorithm may also be a direction for future research. Thus far, perspective-based system decomposition generation was either generated heuristically by researchers or by using existing modularization algorithms aimed only to maximize the modularity-based objective function, without any regards for any particular decomposition perspective. Such an optimization algorithm would aim to find a Pareto-efficient set of modular configurations that satisfies certain rule-based decomposition configuration constraints.

Bibliography

Ahn, J., Choi, M., & Suh, E. S. (2018). Entropy-based system assessment metric for determining architecture's robustness to different stakeholder perspectives. *Systems Engineering, 21*(5), 476-489. doi:10.1002/sys.21448

Baldwin, C. Y., & Clark, K. B. (2006). Modularity in the design of complex engineering systems. In *Complex engineered systems* (pp. 175-205).

Borjesson, F., & Hölttä-Otto, K. (2012). *Improved clustering algorithm for design structure matrix*. Paper presented at the 38th Design Automation Conference, Chicago, Illinois, USA.

Braha, D., Minai, A. A., & Bar-Yam, Y. (2006). *Complex engineered systems: Science meets technology*: Springer Berlin Heidelberg.

Briscoe, G. (2014). *Digital innovation: The hackathon phenomenon* (2052-8604). Retrieved from http://www.creativeworkslondon.org.uk/wp-content/uploads/2013/11/Digital-Innovation-The-Hackathon-Phenomenon1.pdf

Charnes, A., Cooper, W. W., & Rhodes, E. (1978). Measuring the efficiency of decision making units. *European Journal of Operational Research, 2*(6), 429-444. doi:10.1016/0377-2217(78)90138-8

Chief Information Officer, U. S. D. o. D. (2010). The DoDAF architecture framework version 2.02. Retrieved from http://dodcio.defense.gov/Library/DoD-Architecture-Framework/

Clausen, A. C. (1883). United States of America Patent No. 286,531.

Clayton, M. J., Johnson, R. E., Song, Y., & Al-Qawasmi, J. (1998). *A study of information content of as-built drawings for USAA*. Retrieved from http://archone.tamu.edu/crs//documents/publications/asbuilt.pdf

Cooper, W. W., Seiford, L. M., & Zhu, J. (2004). Data envelopment analysis. In *Handbook on data envelopment analysis* (pp. 1-39).

Crawley, E., Cameron, B., & Selva, D. (2015). *System architecture: Strategy and product development for complex systems*: Prentice Hall.

Eppinger, S. D., & Browning, T. R. (2012). *Design structure matrix methods and applications*. Cambridge, Mass.: MIT Press.

Eyres, D. J., & Bruce, G. J. (2012). *Ship construction*: Butterworth-Heinemann.

Frey, D. D., Herder, P. M., Wijnia, Y., Subrahmanian, E., Katsikopoulos, K., & Clausing, D. P. (2008). The Pugh controlled convergence method: Model-based evaluation and implications for design theory. *Research in Engineering Design, 20*(1), 41-58. doi:10.1007/s00163-008-0056-z

Giffin, M., de Weck, O., Bounova, G., Keller, R., Eckert, C., & Clarkson, P. J. (2009). Change propagation analysis in complex technical systems. *Journal* *of Mechanical Design, 131*(8). doi:10.1115/1.3149847

Gu, P., & Sosale, S. (1999). Product modularization for life cycle engineering. *Robotics and Computer-Integrated Manufacturing, 15*(5), 387-401. doi:10.1016/s0736-5845(99)00049-6

Guo, F., & Gershenson, J. K. (2007). Discovering relationships between modularity and cost. *Journal of Intelligent Manufacturing, 18*(1), 143-157. doi:10.1007/s10845-007-0007-y

Hambali, A., Sapuan, S., Ismail, N., & Nukman, Y. (2009). Application of analytical hierarchy process in the design concept selection of automotive composite bumper beam during the conceptual design stage. *Scientific Research and Essays, 4*(4), 198-211.

Hillis, W. D. (2000). United States of America Patent No. 6,097,673.

Hölttä-Otto, K., Chiriac, N. A., Lysy, D., & Suh, E. S. (2012). Comparative analysis of coupling modularity metrics. *Journal of Engineering Design, 23*(10-11), 787-803. doi:10.1080/09544828.2012.701728

Hölttä-Otto, K., Niutanen, V., Eppinger, S., Browning, T. R., Stowe, H. M., Lampinen, R., & Rahardjo, A. (2018). *Design sprint for complex system architecture analysis*. Paper presented at the 30th International Conference on Design Theory and Methodology, Quebec City, Quebec, Canada.

Hölttä, K., Suh, E. S., & de Weck, O. (2005). *Tradeoff between modularity and* *performance for engineered systems and products*. Paper presented at the Proceedings ICED 05, the 15th International Conference on Engineering Design, Melbourne, Australia.

Joo, S.-J. (2009). Scheduling preventive maintenance for modular designed components: A dynamic approach. *European Journal of Operational Research, 192*(2), 512-520. doi:10.1016/j.ejor.2007.09.033

Kim, G., Kwon, Y., Suh, E. S., & Ahn, J. (2016). Analysis of architectural complexity for product family and platform. *Journal of Mechanical Design, 138*(7). doi:10.1115/1.4033504

Kossiakoff, A., & Sweet, W. N. (2005a). Concept definition. In *Systems engineering principles and practice* (pp. 165-193): John Wiley & Sons, Inc.

Kossiakoff, A., & Sweet, W. N. (2005b). Structure of complex systems. In *Systems engineering principles and practice* (pp. 31-49): John Wiley & Sons, Inc.

Okudan, G. E., & Tauhid, S. (2008). Concept selection methods - a literature review from 1980 to 2008. *International Journal of Design Engineering, 1*(3), 243-277. doi:10.1504/ijde.2008.023764

Otto, K. N., & Wood, K. L. (2001). *Product design: Techniques in reverse engineering and new product development*. Upper Saddle River, NJ: Prentice Hall.

Paik, J. K., & Thayamballi, A. K. (2007). *Ship-shaped offshore installations:*

Design, building, and operation: Cambridge University Press.

Pugh, S. (1991). *Total design: Integrated methods for successful product engineering*. Wokingham, England ; Reading, Mass.: Addison-Wesley Pub. Co.

Ramanathan, R. (2003). *An introduction to data envelopment analysis: A tool for performance measurement*: Sage.

Rechtin, E., & Maier, M. W. (2010). *The art of systems architecting* (2 ed.): Taylor & Francis.

Revelle, J. B., Moran, J. W., & Cox, C. A. (1998). *The QFD handbook*: John Wiley & Sons.

Roup, A. V., Bernstein, D. S., Nersesov, S. G., Haddad, W. M., & Chellaboina, V. (2003). Limit cycle analysis of the verge and foliot clock escapement using impulsive differential equations and Poincaré maps. *International Journal of Control, 76*(17), 1685-1698. doi:10.1080/00207170310001632412

Saaty, T. L. (1988). What is the analytic hierarchy process? In *Mathematical models for decision support* (pp. 109-121).

Sanaei, R., Otto, K., Hölttä-Otto, K., & Luo, J. (2015). *Trade-off analysis of system architecture modularity using design structure matrix*. Paper presented at the 41st Design Automation Conference, Boston, Massachusetts, USA.

Schwabl, F. (2006). *Statistical mechanics*. Berlin: Springer.

Sinha, K., & de Weck, O. L. (2013). *Structural complexity quantification for engineered complex systems and implications on system architecture and design*. Paper presented at the 39th Design Automation Conference, Portland, Oregon, USA.

Sinha, K., & de Weck, O. L. (2016). Empirical validation of structural complexity metric and momplexity management for engineering systems. *Systems Engineering, 19*(3), 193-206. doi:10.1002/sys.21356

Sinha, K., & Suh, E. S. (2018). Pareto-optimization of complex system architecture for structural complexity and modularity. *Research in Engineering Design, 29*(1), 123-141. doi:10.1007/s00163-017-0260-9

Sinha, K., Suh, E. S., & de Weck, O. (2018). Integrative complexity: An alternative measure for system modularity. *Journal of Mechanical Design, 140*(5). doi:10.1115/1.4039119

Sosa, M. E., Eppinger, S. D., & Rowles, C. M. (2003). Identifying modular and integrative systems and their impact on design team interactions. *Journal of Mechanical Design, 125*(2), 240-252. doi:10.1115/1.1564074

Sosa, M. E., Eppinger, S. D., & Rowles, C. M. (2004). The misalignment of product architecture and organizational structure in complex product development. *Management Science, 50*(12), 1674-1689.

doi:10.1287/mnsc.1040.0289

Sosa, M. E., Eppinger, S. D., & Rowles, C. M. (2007). A network approach to define modularity of components in complex products. *Journal of Mechanical Design, 129*(11), 1118-1129. doi:10.1115/1.2771182

Suh, E. S., Chiriac, N., & Hölttä-Otto, K. (2015). Seeing complex system through different lenses: Impact of decomposition perspective on system architecture analysis. *Systems Engineering, 18*(3), 229-240. doi:10.1002/sys.21294

Thebeau, R. E. (2001). *Knowledge management of system interfaces and interactions from product development processes.* (Masters Thesis), Massachusetts Institute of Technology,

Ulrich, K. T. (2003). *Product design and development*: Tata McGraw-Hill Education.

Walsh, H. S., Dong, A., & Tumer, I. Y. (2018). *An analysis of modularity as a design rule using network theory*. Paper presented at the 30th International Conference on Design Theory and Methodology, Quebec City, Quebec, Canada.

Yu, T.-L., Yassine, A. A., & Goldberg, D. E. (2007). An information theoretic method for developing modular architectures using genetic algorithms. *Research in Engineering Design, 18*(2), 91-109. doi:10.1007/s00163-007-

0030-1

Zhu, L., Aurum, A., Gorton, I., & Jeffery, R. (2005). Tradeoff and sensitivity analysis in software architecture evaluation using analytic hierarchy process. *Software Quality Journal, 13*(4), 357-375. doi:10.1007/s11219-005-4251-0

Appendix A: Bill of Materials for VFEC Architecture

Verge and Foliot Escapement Clock Architecture Bill of Materials (BOM)
Note: Components B-3, D-6 are unused

Part Number	Diagram	Part Name
1.		(A1) Verge
2.		(A2) Topside Housing
3.		(A3) Wall Mount
4.		(A4) Foliot
5.		(A5) Verge Support
6.		(A6) Topside Struct.
7.		(A7) Coin Basket Pulley Struct.

8.		(A8) Counterweight Support Struct.
9.		(B-1) Main Structure (Back)
10.		(B-2) Main Structure (Center)
11.		(B-4) Second Clock Face
12.		(B-5) Hour Clock Face
13.		(B-6) Main Structure (Front)
14.		(C1) Main Platform
15.		(C2) Second Hand

16.		(C3) Ratchet Pawl
17.		(C4) Pallet
18. 19. 20. 21.		18. Platform Leg 1 19. Platform Leg 2 20. Platform Leg 3 21. Platform Leg 4
22. 23.		(C6/C7) Centrifugal Weight 1 (C6/C7) Centrifugal Weight 2
24.		(C8) Coin Basket
25.		(C9) Ratchet Gear
26.		(C10) Ratchet Gear Cover
27.		(C11) Hour Hand
28. 29.		(C12) Pulley Shaft (Coin Basket) (C12) Pulley Shaft (Counterweight)

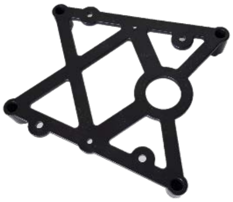
30.		(C13) Thread Controller
31.		(C14) Coin Basket/Hook Interface
32.		(C15) Coin Basket Hook
33. 34.		(C16/C17) Pulley (Coin Basket) (C16/17) Pulley (Counterweight)
35.		(C18/C19) Counterweight
36.		(D-1) Gear
37.		(D-2) Escape Wheel 1
38.		(D-3) Gear

39.		(D-4) Escape Wheel 2
40.		(D-5) Gear
41.		(D-7) Gear
42.		42. 10T Gear
43.		43. 20mm Shaft
44. 45.		31mm Shaft 1 (Hour) 31mm Shaft 2 (Second)
46.		34mm Hex Shaft
47.		45mm Hex Shaft
48. 49.		20cm White Thread 60cm White Thread

Appendix B: Bill of Materials for FPC Architecture

Flying Pendulum Clock Architecture Bill of Materials (BOM)

Part Number	Diagram	Part Name
1. 2.		(A1) Crane Upper Struct. (L) (A2) Crane Upper Struct. (R)
3.		(A3) Ratchet Pawl
4. 5.		(A4) Crane Lower Struct. (L) (A5) Crane Lower Struct. (R)
6.		(B2/A7) Gear Housing
7. 8.		(A7) Ratchet Gear (A8) Ratchet Gear Case

9.		(A9) Large Gear Housing
10.		(B1) Coin Basket Struct.
11.		(B2) Lower Pulley Hinge
12.		(B3) “Weight”
13.		(B4) Black Thread Length Controller
14.		(B5) Coin Basket Struct.
15. 16.		(B6/B7) Upper Pulley Wheel (B6/B7) Lower Pulley Wheel
17.		(B8) White Thread Length Controller

18. 19.		(B9.1) 60 sec Hand (B9.2) 30 min Hand
20.		(B-1) Gear
21.		(B-2) Gear
22.		(B-3) Gear
23.		(B-4) Gear
24.		10T Gear
25.		15mm Shaft
26. 27.		25 mm Shaft (60 sec) 25 mm Shaft (30 min)
28. 29.		40 mm Shaft 1 40 mm Shaft 2
30. 31.		83mm Shaft 1 83mm Shaft 2

32.		110mm Shaft
33.		26mm Hex Shaft
34. 35.		Black Thread White Thread
36.		(C-1) Main Housing
37.		(C-2) 40 mm Shaft Housing
38. 39.		(C-3.1) 60 sec Clock Face (C-3.2) 30 min Clock Face
40.		(C1) Central Column
41. 42. 43. 44.		(C2.1) Column (C2.2) Column (C2.3) Column (C2.4) Column
45.		(C3) Rotating Crane

Appendix C: Bill of Materials for CSC Architecture

Congreve Style Clock Architecture Bill of Materials (BOM)

Part Number	Diagram	Part Name
1.		(A1) Holding Arm L
2.		(A1) Holding Arm R
3. 4. 5.		-(A2) Removable Pin 1 (Top) -(A2) Removable Pin 2 (Bottom L) -(A2) Removable Pin 3 (Bottom R)
6.		(A3) Ratchet Casing
7.		(A4/A6) Shaft Cover
8.		(A5) Time Hand

9.		(A7) Small Pin
10.		(A8) Pin Support (Type 1)
11. 12.		-(A9) Front Pin Support (Type 2) -(A9) Back Pin Support (Type 2)
13.		(A10) Connecting Rod
14. 15. 16. 17. 18. 19.		(A11) Removable Pin 1 (Holding Arm L) (A11) Removable Pin 2 (Holding Arm R) (A11) Removable Pin 3 (Shaft Cover) (A11) Removable Pin 4 (Rod) (A11) Removable Pin 5 (Ball Track-B1) (A11) Removable Pin 6 (Ball Track-B3)
20.		(A12) Coin Basket
21.		(A13) Central Column

22.		(A14) Ratchet
23.		(A15) Shaft Cover
24.		(B1) Main Gear Housing
25.		(B2) Gear Housing Cover
26.		(B3) Structural Supports
27.		Steel Ball

28.		G-1 Gear
29. 30.		G-2 Gear (Upper) G-2 Gear (Lower)
31.		G-3 Gear
32.		G-4 Gear
33.		35mm Shaft
34.		25mm Shaft
35.		20mm Shaft

36.		35mm Hex Shaft
37.		(Part 1) Top Wing Struct.
38.		(Part 2) Time Indicator
39.		(Part 3) Bottom Reinforcement
30.		(Part 4) Ball Track
41.		(Part 5) Crane

Appendix D: DSM for VFEC Architecture

1				1		1										1																																	
2				1		1																																											
3						1			1																																							
4	1	1																			1	1																										
5									1	1							1																																
6	1	1	1						1	1			1																																			
7										1																		1					1																
8										1																			1					1															
9			1		1	1				1				1																																1		
10					1	1	1	1	1				1	1																													1	1	1	1	1		
11													1																																			
12													1																																			
13						1				1	1	1																														1	1	1	1			
14									1	1								1	1	1	1																												
15																																													1				
16																									1																1								
17	1				1																															1		1										
18														1																																		
19														1																																		
20														1																																		
21														1																																		
22				1																																												
23				1																																												
24																														1																		
25																1										1															1					1			1
26																									1																					1			1
27																																												1					
28							1																									1																
29								1																									1															
30																																														1			
31																								1								1																	
32																														1																			1
33							1																				1																					1
34								1																				1																			1	
35																																															1	
36																																					1						1					
37																	1																						1								1		
38																																			1					1		1						
39																	1																				1										1		
40																																										1			1				
41																1									1													1					1				1			
42																																								1	1						1		
43										1			1																									1											
44										1			1														1									1													
45										1			1		1																									1									
46										1			1												1	1				1											1								
47									1	1																											1		1			1							
48																																		1	1														
49																									1	1						1	1																

Appendix E: DSM for FPC Architecture

1		1		1											1																																
2	1				1										1																																
3							1														1																									
4	1				1							1					1																				1								
5		1		1								1					1																				1								
6																				1																1								
7			1					1													1												1		1											
8							1																											1		1	1								
9																		1	1							1	1												1		1	1						
10														1																																
11				1	1										1																												
12																																			1										
13																																			1										1
14										1																										1									
15	1	1																																	1									
16				1	1						1																								1									
17																				1											1													
18									1													1				1																			
19									1																		1																		
20																							1				1									1								
21			1			1	1										1							1									1											
22																		1					1	1		1										1								
23																				1		1			1											1								
24																					1	1										1				1				1				
25																							1													1								
26									1									1				1														1								
27									1										1	1																1								
28																																						1							
29																																						1							
30																																					1								
31																																					1								
32																								1												1	1			1					1
33							1	1									1				1														1								
34												1	1																																1
35							1	1						1	1	1																												
36				1	1	1		1	1										1		1	1	1	1	1	1			1	1	1	1								1	1	1	1	
37																												1	1			1												
38									1																																			
39									1																																			
40																								1								1													1
41																																					1								
42																																					1								
43																																					1								
44																																					1								
45													1																			1		1						1						

Appendix F: DSM for CSC Architecture

1														1																									1	
2															1																								1	
3																							1	1																
4																							1	1																
5																							1	1																
6																					1						1				1									
7																1																				1					
8																																									1
9																								1																
10																	1																						1	
11																		1																					1	
12																			1																				1	
13																1	1																							
14	1																																			1				
15		1																																		1				
16							1						1																											
17										1			1																											
18											1													1																
19												1														1														
20																																									1
21																								1		1															
22						1																						1				1									
23																																				1					
24			1	1	1													1			1				1								1	1	1	1	1		1		
25			1	1	1				1															1								1		1	1	1		1			
26																			1		1																		1		
27																																								1	
28						1																1								1		1	1								
29																														1	1				1						
30																												1	1					1							
31																													1							1					
32						1																1			1			1					1								1
33																								1				1				1									
34																								1	1					1											
35																								1	1				1												
36							1																1	1	1						1										
37														1	1									1																	
38																									1																
39																								1		1															
40	1	1								1	1	1															1														
41								1												1												1									

국문초록

본 연구에서는 시스템 아키텍쳐 설계 단계에서 다양한 이해관계자 관점에 강건한 아키텍쳐를 식별하는 정량적 메트릭을 연구하는 것을 목표로 한다. 복잡도가 높은 시스템 아키텍쳐의 개발 과정에서는 고객 뿐만 아니라 전체 수명주기의 이해관계자의 요구 사항을 감안하여 설계해야 한다. 기술의 발전과 소비자의 요구 사항에 따라서 개발되는 시스템의 복잡도와 수명주기가 지속적으로 증가함에 따라서 시스템의 전반적인 개발, 생산, 운용/유지보수 등 전 단계의 관점을 시스템 기본 설계 단계에서 미리 반영 및 수용을 하는 필요성이 대두되고 있다. 이와 같은 조직적, 기술적인 문제를 해소하는 관점이 어떻게 차이가 나는지 정량적인 평가의 중요성이 부각된다. 통계역학의 엔트로피 기반 메트릭을 개발하여 시스템 분해 관점에 대한 비교를 정량화한다. 개발한 메트릭으로 두 가지 사례연구를 통해서 다양한 관점에 강건한 시스템 아키텍쳐를 평가, 및 전문가의 시스템 분해의 일관성을 평가하는 과정을 거쳐서 제시된 메트릭의 실용성을 분석하였다. 이를 통해서 관점의 차이가 작게 나는 아키텍쳐의 개발에 기여를 하는 데에 목적을 둔다.

주요어: 시스템 아키텍쳐, 시스템 엔지니어링, 모듈화 설계

학번: 2017-29435