A Socio-Technical Systems Analysis of Change Processes in the Design of Flagship Interplanetary Missions

by

Eric D. Ward

Submitted to the Engineering Systems Division in partial fulfillment of the requirements for the degree of Masters of Science in Engineering and Management at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2016

© Massachusetts Institute of Technology 2016. All rights reserved.

Author .................................................................

Engineering Systems Division

May 6, 2016

Certified by .........................................................

Olivier L. deWeck

Professor of Aeronautics and Astronautics and Engineering Systems

Thesis Supervisor

Accepted by .........................................................

Patrick M. Hale

Chairman, Department Committee on Graduate Theses
A Socio-Technical Systems Analysis of Change Processes in the Design of Flagship Interplanetary Missions

by

Eric D. Ward

Submitted to the Engineering Systems Division on May 6, 2016, in partial fulfillment of the requirements for the degree of Masters of Science in Engineering and Management

Abstract

In the engineering of complex systems, changes to flight hardware or software after initial release can have large impacts on project implementation. Even a comparatively small change on an assembly or subsystem can cascade into a significant amount of rework if it propagates through the system. This can happen when a change affects the interfaces with another subsystem, or if it alters the emergent behavior of the system in a significant way, and is especially critical when subsequent work has already been performed utilizing the initial version. These changes can be driven by new or modified requirements leading to changes in scope, design deficiencies discovered during analysis or test, failures during test, and other such mechanisms. In complex system development, changes are managed through engineering change requests (ECRs) that are communicated to affected elements. While the tracking of changes is critical for the ongoing engineering of a complex project, the ECRs can also reveal trends on the system level that could assist with the management of current and future projects.

In an effort to identify systematic trends, this research has analyzed ECRs from two different JPL led space mission projects to classify the change activity and assess change propagation. It employs time analysis of ECR initiation throughout the lifecycle, correlates ECR generators with ECR absorbers, and considers the distribution of ECRs across subsystems. The analyzed projects are the planetary rover mission, Mars Science Laboratory (MSL), and the Earth-orbiting mission, Soil Moisture Active Passive (SMAP).

This analysis has shown that there is some consistency across these projects with regard to which subsystems generate or absorb change. The relationship of the ECR-Subsystem networks identifies subsystems that are absorbers of change and others that are generators of change. For the flight systems, the strongest absorbers of change were found to be avionics and the mechanical structure for the spacecraft bus, and the strongest generators of change were concentrated in the payloads. When this attribute is recognized, project management can attempt to close ECR networks by looking for ways to leverage absorbers and avoid multipliers. Alternatively, in cases where changes to a subsystem are undesirable, knowing whether it is an absorber can
greatly assist with expectations and planning.

This analysis identified some significant differences between the two projects as well. While SMAP followed a relatively well behaved blossom profile across the project, MSL had an avalanche of change leading to the drastic action of re-baselining the launch date. While the official reasoning for the slip of the launch date is based in technical difficulties, the avalanche profile implies that a snowballing of change may have had a significant impact as well. Furthermore, the complexity metrics applied show that MSL has a more complex nature than SMAP, with 269 ECRs in 65 Parent-Child clusters, opposed to 166 in 53 for SMAP, respectively. The Process Complexity metric confirms this, quantitatively measuring the complexity of MSL at 492, compared to 367 for SMAP. These tools and metrics confirm the intuition that MSL, as a planetary rover, is a more complex space mission than SMAP, an earth orbiter.

Thesis Supervisor: Olivier L. deWeck
Title: Professor of Aeronautics and Astronautics and Engineering Systems
Acknowledgments

I would like to thank the Office of the Chief Engineer at the Jet Propulsion Laboratory for their great support throughout this work. I would also like to thank Diego Mundo for his contributions to this research through an internship at the Jet Propulsion Laboratory and through the Undergraduate Research Opportunities Program at the Massachusetts Institute of Technology.
# Contents

1 Introduction .......................................................... 19  
   1.1 Introduction to Change Propagation .......................... 19  
   1.2 Literature Review ........................................... 20  
      1.2.1 Predicting Change Propagation in Complex Design. Clarkson et al., 2001 ........................................ 21  
      1.2.2 Change and customization in complex engineering domains. Claudia Eckert et al., 2003 .......................... 23  
      1.2.3 Change Propagation in Complex Engineering Systems: Monica Griffin et al., 2009 .................................. 27  
      1.2.4 Multilayer Network Model for Analysis and Management of Change Propagation: Michael C. Pasqual and Olivier L. de Weck ................................. 28  
      1.2.5 A Posteriori Design Change Analysis for Complex Engineering Projects: Afreen Siddiqi et al., 2011 ...................... 30  

2 Analysis Methodology .................................................. 33  
   2.1 Timing of Change ............................................... 34  
      2.1.1 Reasons for Change ...................................... 36  
   2.2 Propagation of Change .......................................... 37  
      2.2.1 Absorption and Generation of Change .................. 38  
      2.2.2 Change Propagation Networks ......................... 39  
   2.3 Agents of Change ................................................ 39  
   2.4 Manifestation of Complexity ................................... 41
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>A Comparison of this Methodology to that Proposed in the Literature</td>
<td>42</td>
</tr>
<tr>
<td>2.6</td>
<td>Data Collection</td>
<td>43</td>
</tr>
<tr>
<td>2.7</td>
<td>Data Analysis</td>
<td>49</td>
</tr>
<tr>
<td>3</td>
<td>Soil Moisture Active/Passive</td>
<td>51</td>
</tr>
<tr>
<td>3.1</td>
<td>Overview of the Space Mission</td>
<td>51</td>
</tr>
<tr>
<td>3.2</td>
<td>Timing of Change</td>
<td>53</td>
</tr>
<tr>
<td>3.3</td>
<td>Reason for Change throughout the Project</td>
<td>57</td>
</tr>
<tr>
<td>3.4</td>
<td>Propagation of Change</td>
<td>58</td>
</tr>
<tr>
<td>3.4.1</td>
<td>ECR Parent-Child Networks</td>
<td>60</td>
</tr>
<tr>
<td>3.5</td>
<td>Agents of Change</td>
<td>61</td>
</tr>
<tr>
<td>4</td>
<td>Mars Science Laboratory</td>
<td>65</td>
</tr>
<tr>
<td>4.1</td>
<td>Overview of the Space Mission</td>
<td>65</td>
</tr>
<tr>
<td>4.2</td>
<td>Timing of Change</td>
<td>66</td>
</tr>
<tr>
<td>4.3</td>
<td>Propagation of Change</td>
<td>70</td>
</tr>
<tr>
<td>4.3.1</td>
<td>ECR Parent-Child Networks</td>
<td>72</td>
</tr>
<tr>
<td>4.4</td>
<td>Agents of Change</td>
<td>76</td>
</tr>
<tr>
<td>5</td>
<td>Comparison of SMAP and MSL</td>
<td>81</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of Timing of Change</td>
<td>81</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparison of Propagation of Change</td>
<td>83</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Consistently Strong Absorbers</td>
<td>83</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Consistently Strong Generators</td>
<td>84</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Change Propagation Index</td>
<td>85</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Implications for future missions</td>
<td>85</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparison of Agents of Change</td>
<td>87</td>
</tr>
<tr>
<td>6</td>
<td>Manifestation of Complexity</td>
<td>89</td>
</tr>
<tr>
<td>6.1</td>
<td>Number of Subsystems Affected per ECR</td>
<td>89</td>
</tr>
<tr>
<td>6.2</td>
<td>ECR Parent Child Relationships</td>
<td>90</td>
</tr>
<tr>
<td>6.3</td>
<td>Other Potential Complexity Metrics</td>
<td>92</td>
</tr>
</tbody>
</table>
List of Figures

1-1 Schematic of the multilayer network approach, courtesy Pasqual and deWeck[1] ................................. 20
1-2 Change Prediction Method (CPM), Proposed in Clarkson et al. 2001[2], Courtesy Same. .......................... 21
1-3 Sources of Change Throughout the Design Process, Courtesy Eckert et al.[3] ........................................ 24
1-4 Propagation Behavior, Courtesy Eckert et al.[3] ................. 25
1-5 Example of the CPI spectrum from the complex sensor system project, courtesy Giffin et al.[4] .......................... 28

2-1 Schematic of the Analysis ........................................ 33
2-2 An Example of the Timing of Change Histogram. ................. 35
2-3 Common change timing profiles, proposed in Eckert et al.[3] .......... 37
2-4 Relative Complexity of ECR-to-ECR Network clusters. ............. 42
2-5 Flowchart Schematic of Data Collection .......................... 43

3-1 Soil Moisture Active Passive Satellite, Courtesy NASA[5] ........ 51
3-2 SMAP Satellite and Configurations, Courtesy NASA[5] ............. 52
3-3 Histogram of Change Timing for SMAP ........................... 53
3-4 Cumulative Timing of Change for SMAP .......................... 54
3-5 Cumulative Artifact Releases by Document Type .................... 56
3-6 Reasons for Change across the SMAP Project ..................... 57
3-7 SMAP Change Propagation by Subsystem .......................... 59
3-8 SMAP ECR Parent/Child Networks ............................... 61
3-9 Betweeness Centrality and Weighted Degree of the Engineers that Contributed to SMAP .................................................. 62
3-10 Social Network for the SMAP project as reconstructed from the ECRs 63

4-1 Mars Science Laboratory, Courtesy NASA ................................. 65
4-2 Histogram of Change Timing for MSL ..................................... 67
4-3 Cumulative Timing of Change for MSL ................................. 68
4-4 Cumulative Artifact Releases by Document Type ..................... 69
4-5 MSL Change Propagation by Subsystem ................................. 71
4-6 MSL ECR Parent/Child Networks ......................................... 72
4-7 MSL C1 Cluster .............................................................. 73
4-8 MSL C3 Cluster .............................................................. 74
4-9 MSL C2 Cluster .............................................................. 75
4-10 Betweeness Centrality and Weighted Degree of the Engineers that Contributed to MSL .................................................. 76
4-11 Social Network for the MSL project as reconstructed from the ECRs . 77
4-12 MSL Social Network Before Re-Baselining ............................ 78
4-13 MSL Social Network After Re-Baselining ............................. 79

5-1 Timing of Change Comparison ............................................. 82
5-2 Subsystem Change Propagation Indexes ................................. 86

6-1 Histograms of ECRs by the Number of Subsystems Affected ..... 90
List of Tables

3.1 Document and ECR counts for SMAP .................................. 52
4.1 Document and ECR counts for MSL ................................. 66
5.1 Documents and ECRs of each type for SMAP and MSL ....... 81
5.2 Primary Absorbers of Change ........................................ 83
5.3 Primary Generators of Change ....................................... 84
6.1 Comparison of the Number of ECRs that affect Multiple Subsystems 89
6.2 Cluster Data for ECR Parent-Child Networks .................... 91
6.3 Process Complexity Results ......................................... 94
A.1 SMAP Subsystem Propagation Adjacency Matrix ................. 102
A.2 MSL Subsystem Propagation Adjacency Matrix ................ 103
B.1 SMAP Change Propagation Indexes ............................... 106
B.2 MSL Change Propagation Indexes ................................. 107
Nomenclature

α  Element Complexity

β  Link Complexity

ΔE_{i,j} Binary matrix indicating whether the \( i^{th} \) element is changed because of a change in element \( j \)

\( \mu_i \) Number of Self-Directed ECRs of the \( i^{th} \) subsystem

\( \phi_{ij} \) Number of Edge directions between subsystems i and j

\( C_1 \) Aggregate Element Complexity

\( C_2 \) Aggregate Link Complexity

\( C_3 \) Architectural Complexity

\( CPI \) Change Propagation Index

\( E(x) \) Graph Energy of x

\( n \) number of elements

\( A \) Weighted Adjacency Matrix

ATLO Assembly, Test and Launch Operations

C&DH Command and Data Handling subsystem

CDR Critical Design Review
ChemCam Chemistry & Camera instrument

CPM Change Prediction Method

DCN Design Change Notice, similar documentation to an ECR

DMM Domain Mapping Matrix

DSM Design Structure Matrix

ECR Engineering Change Request

FDD Functional Design Document

FPGA Field-Programmable Gate Array

FS Flight Systems

FSSE Flight Systems System Engineering

GNC Guidance, Navigation and Control subsystem

I&T Integration and Test

ICD Interface Control Document

II&T Instrument Integration and Test

JPL NASA’s Jet Propulsion Laboratory

MOC Mission Operations Center

MSL Mars Science Laboratory, a planetary rover on Mars now called the Curiosity Rover

MTS Mechanical, Thermal and Structures subsystem

NASA National Aeronautics and Space Administration

P-C Parent-Child relationship between ECRs
PDMS  Product Data Management System

PDR  Preliminary Design Review

PFR  Problem Failure Report

PLC  Product Lifecycle

SAM  Sample Analysis at Mars instrument

SE   System Engineering

SIR  System Integration Review

SMAP Soil-Moisture Active/Passive, an earth orbiting satellite that measures ground soil moisture

SOC  Security Operations Center

V&V  Verification and Validation

WBS  Work Breakdown Structure
Chapter 1

Introduction

1.1 Introduction to Change Propagation

To manage changes in complex space system development, modifications to documents are recorded with engineering change requests (ECRs). These documents reflect the scope of work through requirements, the design of the mission, the design of flight and ground equipment, planning for implementation, and the test of flight and ground systems. Hence, documents and ECRs can be regarded as an indicator of the progress of work. Importantly, the propagation of change through the system (as recorded by the ECRs) provides a probe into project behavior.

In an effort to identify systematic trends, this research has analyzed the ECRs from two very different space mission projects led by the Jet Propulsion Laboratory (JPL), in order to classify the change activity and assess change propagation. Four questions were posed to guide this analysis. (1) How closely does the JPL flight development process follow the standard NASA lifecycle? (2) How do changes move from one part of the spacecraft system to another? (3) How do people work together to implement change across the system? And finally, (4) how is complexity manifested through engineering change?

This thesis uses the multilayer network analysis approach to investigate these questions, a schematic of which is shown in figure 1-1. The Change Layer is made up of Engineering Change Records (ECRs) for the space missions, collected from
the electronic databases. The Product Layer is made up of documents and drawings relating to the spacecraft subsystems, as well as some documents specific to system-level areas, such as flight system system engineering. Finally, the Social Layer is made up of the engineers and other professionals that worked together on the design of these spacecraft. The analysis contains aspects of both this multi-layer approach, and some multi-dimensional analyses, insofar that some of the tools and metrics applied combine different dimensions of the data, such as the timing of the changes and the reasons for the changes.

Two projects were investigated in this analysis, in an effort to begin heuristically establishing systematic trends across space missions that could be applied to future efforts. One of the projects was Soil-Moisture Active Passive (SMAP) an Earth orbiting satellite, and the other was the Mars Science Laboratory (MSL) now know as the Curiosity rover.

1.2 Literature Review

There are five primary papers that create the 'knowledge gap' around this analysis. Eckert et al.[3] provide the foundation of the systems engineering analysis through Engineering Change Records, extending the ideas established in Clarkson et al.[2] Giffin et al.,[4] Pasqual and deWeck[1] extend this methodology to analyze a large sensor network, and finally, Siddiqi et al. [6] expand the methodology to stand-alone a posteriori analyses.
1.2.1 Predicting Change Propagation in Complex Design. Clarkson et al., 2001

In 2001, P. John Clarkson, Caroline Simons and Claudia Eckert published "Predicting Change Propagation in Complex Design"[2] at the ASME Design Technical Conference. In this paper, they proposed the idea of – and methodology for – predicting change propagation on a case-by-case basis in large-scale and highly complex systems, using data from GKN Westland Helicopters. They focused on the EH101 platform, a civil and military sea rescue and attack helicopter which had already been in development for 15 years, half of the expected 30 year lifecycle. While the helicopter begins with a baseline design, each customer has different requirements, and thus there are many iterations of the helicopter, often similar to previous versions but almost always unique. Notably, their method thus focuses on the impact of few large incoming changes, and the propagation thereof through the system.

![Diagram of the Change Prediction Method (CPM)](image)

Figure 1-2: Change Prediction Method (CPM), Proposed in Clarkson et al. 2001[2], Courtesy Same.
Change Prediction Method

The authors establish a tool they termed the "Change Prediction Method" (CPM), shown in figure 1-2, that captures past experience about the propagation of change between subsystems—both in terms of likelihood and potential impact—in order to create a model that allows managers and engineers to assess the risk of change early on in the design process. This method begins with a system model based on a decomposition from the designers and managers familiar to the project and its history. The decomposition is recorded in Design Structure Matrices (DSM) which become dependency matrices when populated with the likelihood of a change propagating from one subsystem to another and the severity of the impacts thereof, for each pairwise coupling between subsystems.

The dependency matrices can be combined into a single Product Risk Matrix, which is one of the outputs of the CPM model. The CPM also calculates the cascading change propagation tree from this data, to some reasonably finite step limit, as trail probabilities become very small with growing trail length. This cascading propagation tree is the other primary aspect of the model.

Case-by-Case Analyses

Following the CPM creates a propagation model specific to the system and organization, and Clarkson et al. propose running a simulation (i.e. executing the model, or doing a single analysis) on a case-by-case basis. New product requirements come in from the customer at the chartering of a new variant or due to exogenous changes during development, or from within the organization when problems are found during testing or ideas for improvements are discovered. At the inception of these new requirements the initial area (subsystem) and scope (impact) is recorded, which is then used as an input into the propagation model. The model is able to predict the likelihoods and impacts in each system location, which can be plotted on a Case Risk Plot (showing combined probability on one axis and magnitude of impact on the other, similar to those used in program management in many industries.) With this
additional information, designers and managers are better able to predict the complete impact of the incoming changes and better manage and plan for the upcoming design process.

**Important Points**

Notably, the CPM creates a heuristic model of change propagation for a single system, which is then used to predict the total impact of independent, and known, incoming changes. While there are some systematic insights to be gained from an overview of the model, the paper does not discuss these, choosing instead to focus on the practical aspects of the immediate analysis for a single change.

However, Clarkson et al. sets the stage for thinking about change propagation in large-scale, highly-complex systems, which is foundational for this research.

### 1.2.2 Change and customization in complex engineering domains. Claudia Eckert et al., 2003

In 2003, Claudia Eckert, P. John Clarkson and Winfried Zanker[3] built upon the previous paper, using the same data set as before –the GKN Westland Helicopters’ EH101 platform– but extending the analysis to investigate the implications on a systematic level.

**Sources of Change**

In Section 3.1 and 3.2, Eckert et al. discuss the sources and issues with change, summarized in figure 1-3. Additionally they look into emergent changes in more detail, discussing how problems found during design can often force the design back into a previous phase, at least for the affected sub-systems or components. In section 3.1.2/Problems In Use they point out that many issues can only be established during actual use, claiming that "the ultimate test of any product is of course how it performs in use." This is a particular issue when designing spacecraft, since fixes
to hardware after launch are very difficult (e.g. the Hubble space telescope) or prohibitively expensive, if not impossible.

**Change Propagation Networks**

While formally introduced in the previous paper, section 3.3 provides a more complete treatment of change propagation, and introduces the concept of change propagation networks as another way to visualize the dependency matrices for use in the prediction of changes. They suggest that these networks could be created by applying a fundamental understanding of how the system can be decomposed and how subsystems are linked together, à la the Change Prediction Method. Change Propagation Networks are created with subsystems (nodes) which are connected by propagation links (edges). Eckert et al. propose that these edges are defined by the formal and functional links between the subsystems and the parameters thereof (e.g. Power, Information, Geometry, &c.) that are transmitted between them, since these links create the possibility of change propagation across the interfaces. The change propagation network concept is extended with the idea of propagation behavior, which is shown in figure 1-4. When considering a propagation link, when a change is passed from one subsystem to another, it is defined as being 'propagated' by the former,
and 'absorbed' by the latter. In aggregate, the amount of propagation links out of a subsystem (that over which it could propagate change) is that subsystem's degree of propagation, and the amount of incoming links is its degree of absorption.

Eckert et al. propose four main propagation behaviors: Absorbers, Multipliers, Carriers and Buffers. Absorbers are subsystems that have a higher degree of absorption relative to their degree of propagation, and Multipliers are subsystems with a higher degree of propagation than absorption. As such, Absorbers can absorb more change than they create, thus reducing the overall complexity of the propagation system, while the opposite is true for Multipliers. Carriers are defined as subsystems that have relatively even degrees of propagation and absorption, and thus do not increase to the complexity of the overall problem, but tend to pass changes on to other subsystems while remaining, eg. dimensionally stable, in the case of a simple geometric part. Similarly Buffers have an even degree of absorption and propagation, but at a much higher quantity that Carriers. While this could simply be an artifact of a Carrier that is highly central in the propagation network, Eckert et al. propose that these are subsystems with large margins designed in from the beginning which can absorb a large amount of change while margin remains, but then must propagate these changes to other areas once it runs out. In this case, the propagation behavior
would be time dependent throughout the project, though this facet is not investigated further in their paper. Note that a fifth behavior can also be considered for a sub-system that does not change significantly, termed the Constant. Constants would be expected to have very little participation, if any, in the change propagation network.

**Time Behavior of Change Propagation**

In Section 4.3, Eckert et al. introduce the concept of the time behavior of change, and propose the three standard time profiles shown in figure 2-3, while continuing to look at the change propagation on a case-by-case basis, analyzing the effects of an individual incoming change such as a modification in the customer requirements, or the identification of a manufacturing problem. Once the incoming change is accepted, work begins on the design process and the amount of effort varies as changes are absorbed and multiply across the system along the change propagation network. When the work tapers off and all branches are finally absorbed, the design process is completed for that incoming change, and the time profile in this case is either a ripple or blossom pattern.

The Ripple begins with a large amount of effort that tapers off with time as the scope of work is completed, and Eckert et al suggest that this is a common and optimistic occurrence. The Blossom ramps up to a sustained steady level of work, which finally dies down without getting out of hand, and it is postulated that these are more common when the incoming change is large in impact or scope, especially as compared to incoming changes that spark Ripple processes. Finally, Eckert et al. propose the Avalanche profile, where the design process grows unbounded until significant, often outside, action is taken to bring the process to a close. Avalanche processes are quite problematic, and can be experienced when there are unpredictable propagation effects.
1.2.3 Change Propagation in Complex Engineering Systems: Monica Griffin et al., 2009

Monica Giffin, Olivier L. deWeck, Gergana Bounova, and Rene Keller, along, again, with Claudia Eckert and P. John Clarkson, used the latter author’s previously establish methods to perform an analysis on the engineering design of a complex sensor system. They describe the system as "a large scale sensor system that was developed as part of a billion-dollar class project...on a United States government contract" that contained globally distributed hardware and software subsystems. Giffin et al. were able to obtain all of the approximately 41,500 proposed changes spanning eight calendar years to use as the data set for this analysis.

As opposed to the previous papers—which set out to create a predictive model of change propagation—Giffin et al. used this data set to perform an a posteriori analysis of change propagation in this system, looking at which subsystems had generated or absorbed more or less change over the course of the project. They brought in the metric of the Change Propagation Index (CPI) as a way to quantitatively measure the magnitude of generation and absorption. Equation 1.1 shows the CPI as originally proposed by Suh et al.[7], where N is the number of elements in the system, and E is the binary matrix indicating whether the i\(^{th}\) element is changed because of a change in element j.

\[
CPI_i = \sum_{j=1}^{N} \Delta E_{j,i} - \sum_{k=1}^{N} \Delta E_{i,k} = \Delta E_{out,i} - \Delta E_{in,i} \quad (1.1)
\]

Giffin et al. use the CPI to classify multipliers as elements with \(CPI > 0\), carriers as elements with \(CPI \approx 0\) and generators as those with \(CPI < 0\). This is a similar metric to that proposed by Eckert et al. and shown in figure 1-4, except that the CPI is a one-dimensional metric (as shown in figure 1-5,) and as such, it expresses only the relative proportion of incoming and outgoing change, not the magnitude thereof. As such, for example, CPI cannot distinguish between Constants, Carriers and Buffers.

Finally, Giffin et.al. translated the concept of timing of change behavior profiles
Importantly, Giffin et al. extended the work of the previous papers into the concept of an a posteriori analysis of a previous project, brought the concept of CPI into the fold, and extended the time profiles to include analysing projects as a whole, instead of single instances of incoming change.

1.2.4 Multilayer Network Model for Analysis and Management of Change Propagation: Michael C. Pasqual and Olivier L. de Weck

In 2011 Michael C. Pasqual and Olivier L. de Weck[6] used the same data set as Giffin et al. (the complex sensor system,) and introduced the concept of the multilayer network model, looking specifically at potential tools and metrics that can be used to analyze the model, and how such a model can contribute to the prevention, prediction and control of change propagation.

The model is described largely in two parts. The first half can be considered the individual layer networks: the Product Layer made up of the elements of the system and their relations, the Change Layer made up of the individual changes
and the propagation relationships, and the Social Layer made up of the people who worked on the project and their various relationships, such as theoretical or actual communication links, or chain of command. The second half of the model consists of the inter-layer edges between the aforementioned layers: the Social-to-Change edges showing which engineers worked on which changes, the Change-to-Product edges showing which components were affected by which changes, and the Product-to-Social edges showing which engineers were responsible for designing which elements.

Pasqual and deWeck proceed to define and catalog a repository of tools and metrics that can be used to analyze the multi-layer network. Many pre-existing tools are listed, including the Design Structure Matrix (DSM) and Domain Mapping Matrix (DMM) for analyzing one and two layers at a time, as well as the Change Propagation Index (re-termed the Component Change Propagation Index). They also propose new tools, or adaptations of existing tools for use in this context, such as an extension of the CPI, the Engineer Change Propagation Index, for analyzing the Change-Social dual layer network, and graph theoretic tools to analyze the graph properties of the constituent single-, dual-, and triple-layer networks.

In the case study, Pasqual and deWeck found that, contrary to Eckert et al.’s analysis, change was often isolated, with 91% of un-parented changes having no children. However, their results agreed with Eckert et al. on the finding that change propagation almost always halted after four generations.

Pasqual and deWeck also use the results of their case study to propose that the control of change propagation hinges on the management of the social layer, while acknowledging that there is still much work to be done towards completely understanding this layer’s effects.

Finally, Pasqual and deWeck express that this analysis would best performed during the development effort such that the results could be acted upon, while acknowledging that it is unclear when, or if, there would be a sufficient data available during the course of the project. However, they also articulate the benefits of performing this analysis retrospectively, claiming that there is potential value to future projects since many projects are adaptations of previous projects, or at least anal-
ogous in many ways, and that it may be possible to develop heuristic relationships that will apply to future projects as well.

1.2.5 A Posteriori Design Change Analysis for Complex Engineering Projects: Afreen Siddiqi et al., 2011

In 2011, Afreen Siddiqi, Gergana Bounova, Olivier L. deWreck, Rene Keller and Bob Robinson[1] proposed a range of multi-dimensional analyses of change propagation. This technique differs from the multi-layer approach by eschewing the concept of linking the development layers, in favor of focusing on the process layer from different dimensions, namely the timing of change, location of change (i.e. the system elements that are changing), and cost of change, and the pairwise two dimensional combinations thereof. Change propagation effects were not taken into account for this analysis, and as such, the systematic effects of the locations of change were not analyzed.

The time-only domain analysis is an application of the time-profile analyses introduced by Eckert et al.[3] and extended to a project-wide analysis in Giffin et al.[4] The location-only domain analysis looked at system elements that were hotspots of change activity, and the cost-only domain analysis looked at the characteristics of change cost distributions.

The time-location analysis to split the change into some aggregate unit of time across the project (eg. monthly) to look at the time-based characteristics of the hotspots. Siddiqi et al. proposed that many hotspots of change could be characterized as "early-bloomers" with the bulk of change happening early on in the project, or "late-bloomers" with much of the change activity happening later in the project, with some locations having a uniform distribution of change activity for the duration.

The location-cost analysis attempts to characterize which system elements are cost drivers, postulating that hotspots of change activity may not be drivers of the change cost. Finally, the time-cost analysis characterizes how the costs were incurred over time to identify trends for use in future fiscal management. Note that change propagation effects were not taken into account during this analysis, which can have
effects on the way that total change costs can be assessed.

This aggregated analysis was performed over the case study of the BP Angola Block 18 Greater Plutonia development project, using 1147 design change notice (DCN) records spanning the 4 year project. The DCNs included data on whether the change was approved or rejected (allowing the calculation of the Change Acceptance Index), the estimated cost, final cost (if approved), as well as the individuals contributing to the DCN, which were used to approximate the system elements affected by each DCN. The DCN also contained the other relevant data for the analyses, including those points that are considered fairly standard for engineering change records across the industry (eg. relevant dates.)
Chapter 2

Analysis Methodology

In chapter 1, the constituent parts of a system have been referred to as the system *elements*, which allows for the extension of the models and analyses across a wide range of different systems. However, in the case of JPL-led space missions, the system can be defined more specifically, in this case, the system is the spacecraft product designed through the engineering process, and the elements are the spacecraft *sub-systems* (such as the Instrument Antenna, or Flight Software Avionics) and will be referred to as such for the duration of this analysis.

Engineering Change Requests give us a lens through which we can look at many

![Figure 2-1: Schematic of the Analysis](image-url)
different aspects of the design process. The data available through Engineering Change Request (ECR) records directly supports analyses that reveal characteristics of the project, namely the first three questions asked in section 1.1: (1) the overall timing of the project and the engineering effort involved, (2) how changes propagate through the system, and (3) who are the agents of these changes.

Through these direct analyses we can gain some perspective on the more elusive characteristic of complexity. Each of the three direct analyses show components that relate to the complexity of the system, especially in regards to the apparent complexity, as it is noticed by the people designing and interacting with the system.

The Propagation of Changes reveals how interlinked the subsystems are, and which have more or less of a role in the overall complexity of the system. The Timing of Changes shows the when of change propagation, revealing the delays in identifying the impacts of changes, which can have a very large impact on the project schedule and cost. Finally, the Agents of Change analysis not only shows us how the contributors work together, but also how the organization anticipates these changes. Although this directly indicates not the complexity itself, but the perception thereof, this perception is an important aspect of complexity and is often called the "apparent" complexity. These factors, and how they relate to complexity are considered during each of the three direct analyses. Finally, this analysis will take a direct look at the complexity of the system, focusing on the latent change propagation as a window to the inherent complexity that is not necessarily apparent at first pass, as demonstrated in figure 2-1—a simple schematic of the analysis.

2.1 Timing of Change

The NASA design lifecycle is split into five primary phases.[8] During Phase A the mission concept is developed and requirements are identified. Phase B sees preliminary design and implementation planning. Preliminary Design Review (PDR) is a milestone review intended to confirm that the preliminary design is complete before proceeding to the detailed design work and fabrication of Phase C. Similarly, Critical
Design Review (CDR) is a milestone review in Phase C where it is shown that the detailed design is complete and the system is ready to be built and integrated. The System Integration Review (SIR) marks the beginning of Phase D. Phase E begins shortly after launch, when the systems is in place and ready to begin its operational life. Within the purview of engineering change analysis, Phase E marks the end of a project.

For this analysis, as mentioned in section 2.6, documents include all of the artifacts classified as documents, or 'doc', in PDMS. Added to these documents are the artifacts classified as engineering drawings, or 'dwg', which are specifically referenced by an ECR. Most drawings are excluded due to the fact that the changes to these artifacts are usually controlled through a process separate to the ECRs. As such, to includ all of the drawings would skew the time metrics of the documents and ECRs. Therefore, drawings are only included when they are specifically referenced by an ECR.

The Timing of Change analysis aggregates the initiation and release of ECRs and Documents into a histogram representation across the project. The data points are split into sections of 60 days, and mild smoothing is applied to the resulting histogram to assist with legibility. Specifically, a bi-directional smoothing was applied with a smoothing factor $\alpha = 0.9$ over 10 iterations, termed vars.alpha and vars.exponiters in the code, respectively. This smoothing regime was found to reduce the level of noise without altering the overall profile of the timing data.

Figure 2-2 shows an example of the overall time profile -SMAP in this case- as a histogram of the engineering activity, including the initial releases of documents and the initiation dates of the ECRs, along with phase and milestone review dates. These plots are created for each analyzed project, which are then be compared to the time behavior profiles from Eckert et al.[3], and investigated for aberrant behaviors. These standard
profiles are the Ripple, Blossom and Avalanche, as shown in figure 2-3.

Ripple projects begin with a significant amount of change requiring a lot of initial effort that tapers off over time and is considered to be a healthy and optimistic occurrence. Many planners and participants assuming this case, at least for re-design processes resulting from a single incoming change, as discussed in Eckert et al.[3] This is the time profile observed by Siddiqui et al.[6] for the BP Angola Block 18 Greater Plutonia project, except that they found the highest peak to be late in the project, and termed that a 'late ripple', as opposed to the 'early ripple' shown in figure 2-3.

Blossom projects begin with moderate effort that rises quickly as the full scope of the project is identified but then plateaus while the work is completed and finally tapers off, never having gotten out of hand. Projects exhibiting the Blossom profile are also considered to be healthy occurrences, especially when seen in the context of the full project lifecycle, rather than a single instance of incoming change. For example, many development projects are projected to have a staffing plan similar to the blossom, with a small number of engineers early on during the architecting phase, a high—but steady—staff during the bulk of the design phase, and then maintaining only a small number of engineers to work with the manufacturing staff during the final stage of the project.

Finally, projects Avalanche when the design process grows unbounded until drastic action is taken to bring the work under control. Avalanches of work in a project can be quite problematic, and can be experienced when there are unpredictable complexities in a project that increase the propagation of change beyond the expected amount, or create uncertainties with regards to design decisions.

2.1.1 Reasons for Change

SMAP had an additional data set recorded with each ECR on the Reason for the initiation of the ECR, listed as a choice of one of twelve options. While Eckert et al.[3] discuss the sources of change, this data can be used to investigate how these sources manifest. This data allows for two multi-dimensional analyses of the reasons for change across for the SMAP project that are not possible with the data from
There were 12 different reasons listed, which were grouped into four main categories. The Discovery category is comprised of 'Definition' and 'Scope' reasons, and describes changes made due to the discovery of mistakes or inaccuracies. Clarity is just the 'Clarification' reason, which was by far the most cited reason on ECRs for SMAP. Interfacing is comprised of 'Compatibility', 'Interface Control', and 'Operations' reasons which describe ECRs initiated in order to change the various interfaces between subsystems. Finally, Improvements, comprised of 'Design Improvement' and 'Risk Improvement', describe situations where the engineer has identified changes that can improve on the design, which are primarily internally instigated changes.

2.2 Propagation of Change

Often, in the development of complex systems, change can propagate from one subsystem to another. When a problem is identified in one area, it may take changes across multiple areas to arrive at a satisfactory solution that maintains appropriate system behavior and meets high-level requirements. Pasqual and de Weck [1] eloquently describe this process by explaining that "change propagation occurs when making a single change ultimately requires the implementation of multiple changes in order to achieve the objective of the intended redesign."
2.2.1 Absorption and Generation of Change

In order to investigate the propagation of change through the ECRs, each ECR was correlated to the subsystem that caused the change and the subsystems that were affected by it, revealing intra-ECR links between subsystems. Since the ECR records themselves did not clearly identify subsystems, the functional role of the ECR initiator was used as the location of the cause of the change, and the roles of the authors of the affected documents as the location of the effects of the change. Most of these roles map directly onto flight elements –such as the FS Avionics– though some map to more wide-reaching job functions –such as flight system systems engineer (FSSE)– as discussed in section 2.6.

For clarity, the directedness of change is explicitly defined as follows:

- **Self-directed** changes are defined as changes where all of the affected documents are in the same subsystem as the ECR.
- A subsystem is considered the ‘generating’ subsystem when it is the initiator of an ECR that affects at least one different subsystem.
- Similarly, a subsystem is counted as an ‘absorbing’ subsystem when a document from that subsystem is affected by an ECR that was generated in a different subsystem. Thus, a single ECR could generate change that is absorbed by multiple subsystems.

With these aspects defined, multiple metrics can be assessed for the project. The number of self-directed changes is determined for each subsystem, and shows which subsystems manifest more or less change within them throughout the design process. This is similar to the ‘hotspots’ analysis proposed in Siddiqi et al.,[6] except that Siddiqi et al. do not consider change propagation, and in this case propagated changes are specifically excluded. This allows for the hotspots to be identified in a more focused manner, while avoiding convolution with the effects of change propagation. In this case then, a hotspot is now defined as a subsystem that requires a significant amount of change within the subsystem itself, across the duration of a project.

Additionally, the number of ECRs that each subsystem generates, and the number
of ECRs that each subsystem absorbs, is calculated and rank-ordered. This shows the relative magnitudes of the degree of absorption and degree of propagation (as defined by Eckert et al.[3]) for each subsystem. This data is then used to categorize which subsystems were the primary Absorbers, and which were primary Generators, based on their relative quantity of change propagation in comparison to the other subsystems in that project.

2.2.2 Change Propagation Networks

Eckert et al. propose that these networks are defined by the formal and functional links between the subsystems and the parameters thereof (eg. Power, Information, Geometry, &c.) that are transmitted between them, since these links create the possibility of change propagation across the interfaces. However, this analysis extends the concept to analytic networks based on heuristic propagation data between subsystems, for this a posteriori perspective.

Change propagation networks are created with ECRs (nodes) which are connected by Parent-Child links (edges). When an ECR does not fully fix the issue that was originally identified, an additional ECR is initiated which links to first in a Parent-Child (P-C) relationship. These ECR networks can grow and combine over the course of the project, revealing not only the apparent complexity of the design process, but also the structural complexity of the product that allows change to flow between areas.

2.3 Agents of Change

It is important to understand the network of individuals who carry out the work to effect changes across the system during design. These relationships create a network that describes how the people manage change and work together to control the system.

In order to apply graph theoretic tools and metrics to the social layer, the network is assembled from the data set for the project. In this network the people are located at the nodes, and the edges are the connections between ECRs initiators and the
author(s) of the documents those ECRs affect. So when a person initiates an ECR their node would be connected to that of a person who authors a document that ECR affects.

The natural clustering of the network is then assessed using the Louvain modularity algorithm[9]. This modularity metric naturally groups the nodes together based on their connectedness, prioritizing groupings that concentrate connections within the groups while minimizing the connections that link to nodes outside of the group.

Once the social network has been constructed, techniques from graph theory can be used to analyze the relationships. Since the relationships are defined by the ECRs, and two people can work on multiple ECRs together, the links are weighted by the number of ECRs that those two people worked on together (i.e. these are not strictly binary relationships.) The "degree" of a node is simply the number of links that connect to that node, and in this network we can consider the 'weighted degree,' which counts not just the presence of a link, but the weight of each link as well (i.e. the sum of the weights of all the links that connect to that node.) As such, weighted degree shows how prolifically connected a node is, and in this network becomes simply a count of how many ECRs that individual worked on (with another person.)

Another applicable graph theoretical metric is 'betweenness centrality,' which is a measure of how essential a node is in connecting people across the network. For each arbitrary pair of nodes there may be many paths from one to the other traveling from node to node along the edges. The 'shortest path' is the single such path that passes through the fewest nodes, as compared to the other valid paths between that pair. The betweenness centrality of a node is the number of shortest paths for the whole network that pass through that node.

These graph-theoretic tools and metrics facilitate an understanding of the social network that can reveal multiple insights to its organization.
2.4 Manifestation of Complexity

Two potential measures of complexity are postulated. The first is related to the number of ECRs that affect multiple subsystems, and the second is related to the number of ECRs that spawn additional ECRs in "parent-child" clusters. In the first case the relationship is identified at the time the ECR is initiated, whereas in the second case there is a delay in the realization that the change has propagated, resulting in another ECR initiated later in time. This later metric is particularly interesting, since it combines the timing of change (in this case changes that evolve slowly, and thus have a greater potential to cause avalanches) with the propagation of change, and this could be considered to be one aspect of a multi-dimensional analysis.

When an ECR is created in response to a previous change, also recorded in an ECR, it creates a parent-child relationship that can also be mapped into a network. Sometimes a single ECR may have multiple children (considered siblings to each other) and a child ECR may spawn more itself. Furthermore, chains and clusters can originate independently, but be connected later by a child ECR that references multiple parents, each from different clusters.

These relationships reveal a different level of complexity than direct ECR-subsystem relationships, since (some of) the effects of the parent ECR were not considered during its tenure and needed a separate ECR to be created at a later date. This distinction highlights the human aspects of dealing with complex systems, and reveals the complexity of the system.

Figure 2-4 shows a number of different cluster configurations found in SMAP, and it is postulated that complexity in cluster geometry relates to complexity in project implementation.
2.5 A Comparison of this Methodology to that Proposed in the Literature

Similar to the more contemporary papers in the literature review, this analysis considers the project as a whole, rather than using the tools and metrics to analyze a single instance of incoming change as originally proposed by Eckert et al.[3] As such, rather than to plan for short-term development efforts, this analysis attempts to establish systematic trends across multiple projects with the aim that the insights gained can be applied to future projects, as well as informing future changes to the organization or design process.

Unique to this analysis is the consideration of both the multilayer and multidimensional perspectives. This allows the analysis of some of the more complicated interactions across a project that consider multiple layers with a second dimension, such as how the way that engineers (social layer) manage change (process layer) changes throughout the timing of the project (additional dimension).

Finally, the data set available for this analysis also puts some constraints on which tools and metrics can be applied. For example, costing data was not available with the ECRs, which excludes the possibility of analyzing the cost dimension. Additionally, as mentioned in section 2.6 the Work Breakdown Structure (WBS) was used to associate the engineers with the subsystems they worked on (i.e. the social layer and the
product layer,) which precludes analyses such as investigating how these roles change over time.

2.6 Data Collection

The data was collected from the electronic databases of the projects, in this case, the Product Data Management System (PDMS) at the JPL. Both Engineering Change Records (ECRs) and Documents (as well as the Drawings that were referenced by ECRs) were collected into a MatLab database. Additionally, the Work Breakdown Structure (WBS) for the projects were collected from the respective Program Managers and used to correlate the Documents with the system elements (subsystems) that the documents referenced. Finally, the project milestones were collected from the JPL intra-net. This data collection resulted in four input databases used in the analyses.

Figure 2-5 shows a simple schematic of the data collection process, with data sources shown in green and Matlab tables shown in orange, which are inputs to the Analysis. Note that the MasterRoles table is not a direct input to in the Analysis, but is used before the analysis to fill the missing information in the MasterECR and MasterDoc tables.

Figure 2-5: Flowchart Schematic of Data Collection
ECR Table

The data from the ECRs were collected in the MasterECR Matlab table, which contains the following information:

- **ECR Number (ECRNumber):** the six digit unique identifier assigned to each ECR. This is recorded as a six digit string.
- **ECR Reason (ECRReason):** the reason given for initiating the change, recorded on the ECR form, when available. This is recorded as one of the following strings:
  - Clarification
  - Compatibility
  - Cost Reduction
  - Deficiency
  - Design/Risk Improvement
  - Interface
  - Production (schedule)
  - Operational
  - Resources (Mass/Power/Memory)
  - Safety
  - Scope
  - Vendor Driven
- **Record List (RecordList):** A list of the associated records (documents and drawings) that were affected by the ECR. This is recorded as a cell array where each cell contains a string of the unique identifier of one of those documents.
- **Unknown Records:** A list of the associated records that were unrecognized, but recorded to be affected by the ECR, often empty. This is recorded as a cell array where each cell contains a string of the unique identifier of one of those documents.
- **ECR Subsystem:** The subsystem where this change began. This was filled by correlating the individual who initiated the ECR with his or her functional role –using the WBS– and assigning the subsystem associated with that functional role as the ECR subsystem.\(^1\) This is recorded as one of the same strings as the

\(^1\)As a side effect of this, some ECRs (and documents) have assigned subsystems that are not
level four subsystem entries in the roles table.

- Initiated Date: The date that the ECR was initiated, used as the benchmark date for the timing of the ECR and recorded as a Matlab datetime.
- ECR Initiator: The anonymized identifier of the individual who initiated the ECR, recorded as a string.

The row names in the Matlab table are the ECR numbers, and the variable names are listed in parentheses above.

**Document Table**

The data from the documents were collected in the MasterDoc Matlab table. This included traditional documents such as those defining the system requirements or procedures, as well as engineering drawings that were specifically referenced by one or more ECRs. However, most of the engineering drawings were controlled through a separate process, and as such their complete inclusion in the data set (particularly with regards to the timing analysis) would have created erroneous artifacts that do not relate to the engineering change records.

The MasterDoc table contained the following items for each document:

- **Document Number (DocNumber):** The unique identifier of each document, as assigned in PDMS. Recorded as a string.
- **Release Date (ReleaseDate):** The initial release date of the document, used as the baseline for the timing of document releases, recorded as a Matlab datetime.
- **Type (Type):** The Document type, as specified in PDMS, recorded as one of the following strings:
  - Agreement/Contract
  - Design
  - Dictionary
  - Drawing
  - Engineering

traditionally associated with subsystems. However, this does allow the possibility of looking at system-level areas, such as flight system system engineering.
- Guideline
- Interface Control
- Lists
- Matrix
- Operations
- Plan
- Policy
- Procedure
- Program
- Proposal/Announcement
- Requirement
- Specification
- Test
- Test Report
- Unspecified

- Document Subsystem (DocSubsys): The subsystem associated with this document. Similarly to ECRs, this was filled by correlating the individual who was the assigned author of the document with his or her functional role, and assigning the subsystem associated with that functional role as the Document subsystem. This is recorded as one of the same strings as the level four subsystem entries in the roles table.

- Author (Author): The unique, anonymized, identifier of the individual assigned as the author of this document, which are, of course, the same identifiers for a single individual across the project, whether as a document author or an ECR initiator.

The row names in the Matlab table are the document numbers, and the variable names are listed in parentheses above.

Roles Table

The data from the WBS is cataloged in the Matlab MasterRoles table, which merely correlates an individual with his or her level 3, and level 4 subsystem. This table includes the following data for each individual:

\(^2\)used only when the document had no specified type in PDMS.
- Unique Identifier (Username): This is the unique, anonymized, identifier for each individual.
- Level 3 Subsystem (SubsystemL3): This is the level three subsystem of the individual, recorded as one of the following strings:
  - Business Office
  - EPO
  - Instrument
  - Launch Vehicle
  - Mission Assurance
  - Mission System
  - Project Management
  - Project SE
  - Spacecraft
  - SRB
  - Unknown

Note that while Business Office, EPO and SRB were all valid level three roles, none of these were encountered on an individual who initiated an ECR, or authored a document.

- Level 4 Subsystem (SubsystemL4): This is the level four subsystem, getting one level more specific than the level three subsystem. It is recorded as one of the following strings:
### MSL vs. SMAP

<table>
<thead>
<tr>
<th>MSL</th>
<th>SMAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS Avionics</td>
<td>Spacecraft / C&amp;DH</td>
</tr>
<tr>
<td>Flight Software</td>
<td></td>
</tr>
<tr>
<td>FS GNC</td>
<td>Spacecraft / GNC</td>
</tr>
<tr>
<td>FS Mechanical System</td>
<td>Spacecraft / Structure</td>
</tr>
<tr>
<td>FS Power</td>
<td>Spacecraft / Power</td>
</tr>
<tr>
<td>FS Propulsion</td>
<td>Spacecraft / Propulsion</td>
</tr>
<tr>
<td>FS System Engineering</td>
<td>Spacecraft / ATLO</td>
</tr>
<tr>
<td>FS Telecom</td>
<td>Spacecraft / Telecom</td>
</tr>
<tr>
<td>FS Thermal Control</td>
<td>Spacecraft / Thermal</td>
</tr>
<tr>
<td>Ground Data System Mgmt.</td>
<td>Mission System / SOC</td>
</tr>
<tr>
<td>Mission Assurance</td>
<td>Mission Assurance</td>
</tr>
<tr>
<td>Mission Design</td>
<td>Mission System / SE</td>
</tr>
<tr>
<td>Mission Operations System</td>
<td>Mission System / MOC</td>
</tr>
<tr>
<td>Payloads</td>
<td>Instrument / Antenna</td>
</tr>
<tr>
<td>Project Mgmt</td>
<td>Project Mgmt.</td>
</tr>
<tr>
<td>Project Science</td>
<td>Mission System / SOC</td>
</tr>
<tr>
<td>Project System Eng.</td>
<td>Project SE</td>
</tr>
<tr>
<td>Sample Acquisition</td>
<td></td>
</tr>
<tr>
<td>System Integration and Test</td>
<td>Spacecraft / I&amp;T</td>
</tr>
<tr>
<td></td>
<td>Instrument / I&amp;T</td>
</tr>
<tr>
<td></td>
<td>Instrument / MTS</td>
</tr>
<tr>
<td></td>
<td>Instrument / Radar</td>
</tr>
<tr>
<td></td>
<td>Instrument / Radiometer</td>
</tr>
<tr>
<td></td>
<td>Launch Vehicle</td>
</tr>
<tr>
<td></td>
<td>Mission System / Risk Mgmt.</td>
</tr>
<tr>
<td></td>
<td>Spacecraft / Mgmt.</td>
</tr>
<tr>
<td>Unknown</td>
<td>Unknown</td>
</tr>
</tbody>
</table>

The list above correlates subsystems between SMAP and MSL. MSL only has one level four subsystem without a direct equivalent in SMAP (Sample Acquisition), while SMAP goes into greater decomposition in some areas and thus has multiple level four subsystems without direct equivalents in MSL. Note that while some of these strings may vary from project to project, it is critical that a single subsystem is always referred to by the same name for all entries of a single project.

The row names in the Matlab table are the unique identifiers, and the variable names are listed in parentheses above.
Project Milestones Table

Finally, the project milestones were collected in a Matlab table, which contains the following data for each milestone:

- **Name (Name):** The name of the milestone, containing at least the following two milestones, recorded as one of the following strings:
  - Phase A*
  - Phase B
  - Phase C
  - Phase D
  - Phase E*
  - PDR
  - CDR
  - SIR
  - LAUNCH

Items marked with an asterisk are critical to the analysis, since they define the range of design across the project. MSL had a unique project structure due to the re-baselining event, and contained SIR1, SIR2, SIR3 and REBASELINE milestones as well.

- **Date (Date):** The date of the milestone, recorded as a Matlab datetime.

The row names in the Matlab table are the milestone names, and the variable names are listed in parentheses above.

2.7 Data Analysis

The four tables listed in section 2.6 in addition to the project identifier used in PDMS (SMAP and MSL, in this case) recorded as a string with the name 'proj' were used as the input to the analysis code in Matlab. This code is not listed as an appendix due to size limitations. Once the tables are correctly loaded, the FullAnalysis.m master script will run through all of the analyses for the project, which are described in detail in chapter 2.
Chapter 3

Soil Moisture Active/Passive

3.1 Overview of the Space Mission

Soil Moisture Active/Passive (SMAP), shown in figure 3-1, is an earth orbiting satellite with a single scientific mission and two primary instruments. The primary mission is to observe soil moisture and freeze/thaw state from space, and is comprised of the following five goals:[5]

- Understand processes that link the terrestrial water, energy, and carbon cycles,
- Estimate global water and energy fluxes at the land surface,
- Quantify net carbon flux in boreal landscapes,
- Enhance weather and climate forecast skill, and
- Develop improved flood prediction and drought-monitoring capability.

SMAP was launched early November 2014 with commissioning in late January 2015, after just over six years of development beginning with the start of Phase A in June 2008, and has a planned end of mission in early 2018 after 36 months of science observation. SMAP gathers data with two co-located instruments—an active radiometer
and passive radar—via a six meter conically-scanning reflector antenna that spins at up to 14.6 rpm, reflecting the signal from 35.5° off nadir. Figure 3-2 depicts the Spacecraft Bus and Instrument Antenna, and demonstrates the various configurations across the mission, before and after the antenna deployment.

As a general magnitude of the scope of the design, Table 3.1 shows the number of Documents and ECRs of various types for the SMAP project. The SMAP project generated 991 Documents, including engineering drawings that were specifically referenced by an ECR, and a total of 570 ECRs.

<table>
<thead>
<tr>
<th>SMAP</th>
<th>Docs.</th>
<th>ECR</th>
<th>ECR/Doc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>193</td>
<td>353</td>
<td>1.89</td>
</tr>
<tr>
<td>Engineering Design</td>
<td>102</td>
<td>127</td>
<td>1.26</td>
</tr>
<tr>
<td>Plans or Procedures</td>
<td>653</td>
<td>30</td>
<td>0.05</td>
</tr>
<tr>
<td>Operations</td>
<td>25</td>
<td>112</td>
<td>4.48</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>991</strong></td>
<td><strong>570</strong></td>
<td><strong>0.58</strong></td>
</tr>
</tbody>
</table>

1 Documents and ECRs of Unspecified type are not explicitly shown.
2 Some ECRs affect more than one document type.
3.2 Timing of Change

Figure 3-3: Histogram of Change Timing for SMAP

Figure 3-3 shows the timing of change analysis for the SMAP project. SMAP’s document generation builds through Phase B to about 40 documents per 60 days, then remains fairly constant through CDR. Prior to SIR there is another spurt in document generation to a maximum of about 55 documents per 60 days around SIR, then a tapering off towards launch. The ECRs show roughly the same profile. It can be clearly seen in figure 3-3 that the ECRs (as expected) lagged behind the initial document releases.

The plateau of change between PDR and CDR (and the preceding acceleration) can be expected based on the NASA Product Lifecycle[8] (PLC), since this is the time that the bulk of the detailed design work should be happening. SMAP shows the normal tapering off of work after SIR since, being an Earth-orbiting satellite, launch windows occurred often.

The overall profile is a classic "blossom" pattern, where the project begins slowly,
scales up quickly to a plateau, and then tapers off. This is generally considered to be a healthy time profile and mirrors the expected engineering workloads described in the PLC.

Figure 3-4 shows the documents and ECRs plotted cumulatively, and on average SMAP had 0.73 ECRs per document. Figure 3-5 shows the same, but separated by the document type, and the ECRs that affect a document of that type. For requirements artifacts, where the most change is documented, ECRs outnumber documents by 2:1 for SMAP, and continued well into Phase D. This shows that many requirements were being changed instead of waived when not met, with the systems engineers using the low-level requirements to re-allocate and balance margins. Note that this ECR count is based on the number of times a Document of each type is changed by an ECR, such that a single ECR can count towards multiple types, or even multiple counts of the same type. As such, the total numbers differ from table 3.1.

The cumulative time distribution of Engineering artifacts –figure 3-5b– shows a clear acceleration in the rate of ECRs for this document type during Phase C, with
the rate tapering off during Phase D, and, as expected, nearing zero as launch is approached. This profile is expected based on the NASA Product Lifecycle, with the ECRs that were initiated post-SIR being mostly fixes to issues identified during manufacturing and testing.

The cumulative time distribution of Operations artifacts—figure 3-5d—shows that most of these ECRs, and many of these documents weren’t generated until after CDR. Since these artifacts guide the procedures used while the spacecraft is on orbit, it makes sense that the construction and revision of these would wait until the bulk of the design was complete, to insure accuracy to the final product.

Even though SMAP exhibits a relatively well behaved blossom, there are some late changes in the project shown by the ECR activity after SIR. This is in contradiction with the traditional notion that the design should be frozen at CDR. However, in practice issues are identified during manufacturing, integration and testing that require additional changes, even if the initial design process has been completed. The change activity broken out by type in figure 3-5 also elucidates the fact that many of the late project changes are made to Operations type documents, which are intended to be back loaded in the project so that they will be accurate to the final design.
Figure 3-5: Cumulative Artifact Releases by Document Type
3.3 Reason for Change throughout the Project

SMAP engineers provided consistently recorded reasons for each ECR in the documentation, with a fair portion of ECRs citing multiple reasons. The first reason analysis is in the Reason-by-Time domain, shown in figure 3-6a. This analysis looks at how the reasons for needing to instigate change vary across the duration of the project. It can be seen that, expectedly, the aggregate profile is consistent with the
overall blossom shape of the timing of change, however, there is some variability in
the particular reasons. Notably, the Discovery group is more back-loaded, peaking
after CDR and into Phase D. While it would be expected that much of the intentional
discovery and scoping activities would be carried out early in the project, it makes
sense that changes to reduce scope, and make definitions more clear and consistent
would take place after CDR, when the project needs to be tracking towards com-
pletion. Additionally, ECRs citing the Clarity reason peaked before CDR, which is
consistent with the expectations of work priority across a project; when de-prioritizing
non-critical updates during time critical phases, engineers might abstain from some
unnecessary clarification changes.

The second reason analysis is in the Reason-Subsystem domain. Figure 3-6b shows
the reason types separated by the directedness of the change, i.e. whether the change
affects only the same subsystem in which is was initiated, or if the change initiated in
one subsystem affects a different subsystem. The Clarification and Discovery reasons
have a similar proportion of self-directed to outwardly directed changes, at 0.53:1 and
0.51:1, respectively. ECRs initiated for Interface reasons have a much higher ratio
of outwardly directed changes, 0.96:1, which can be expected by the intrinsically
cross-subsystem nature of interfaces. Finally, Improvements also have a high ratio of
outwardly directed change at 0.82:1, although the cause for this is not as clear as it
is for Interfaces reasons. It is postulated that this high proportion of Improvement
changes that link subsystems is due in part how integrated the system is, in this case,
implying high integration/low modularity, such that an improvement would tend to
affect multiple subsystems.

3.4 Propagation of Change

The Propagation of Change data is presented in figure 3-7, which looks at the process
layer with the propagation-by-location dimensions.

The largest sources of self-directed change in SMAP are the system engineering
groups (Spacecraft/FSSE and Project SE, as well as the subsystem SE groups), having
Figure 3-7: SMAP Change Propagation by Subsystem
initiated over 160 ECRs that do not affect other subsystems, which is significantly more than the second largest self-directing subsystem. They also initiated the most ECRs that affect other systems (figure 3-7b) and were the largest absorber of change as well (figure 3-7c). This shows that system engineers are more than just a hotspot, but a locus of change. While they alter their own documents more often than not, these subsystems generate and absorb a lot of change to and from the rest of the system, which goes beyond merely being a hotspot that contains a lot of change within.

The top two generators of change, after systems engineering, are the ATLO activities and the payload (Instrument Antenna for SMAP.) The payloads set interface requirements on the spacecraft system while the integration and test activity uncovers issues at the system level that requires modifications.

Many of the main absorbers were expected, such as Command and Data Handling (C&DH.) Surprisingly, however, the structural subsystems – Instrument Mechanical, Thermal and Structural (MTS), and Spacecraft Structure– were significant absorbers. MSL shows a similar pattern, and will be discussed further in section 5.2.

3.4.1 ECR Parent-Child Networks

It is also possible to analyze the propagation-by-time dimension. The majority of ECRs in the SMAP project were not part of inter-ECR networks (447 out of 570,) which is consistent with the finding by Pasqual and deWecK[1], though conflicting with Eckert et al.[3]. The SMAP ECRs that were part of parent-child clusters are shown in figure 3-8, and the ECRs are colored based on the time they were initiated, with red signifying the beginning of the project through purple at the end. The majority of the clusters are simple linear or branching chains, and most of the clusters have two or three members, but there are multiple clusters with every member count up to eight.

Beyond these clusters, there is a single cluster of 21 ECRs which stands out, not only because of its size, but by the fact that the network spans a very large portion of the project, almost one and a half years. This is related to the power system, including
solar arrays, pyros for deploying the system, switches, etc. The initiating ECR in this cluster occurred early in the project when it was identified that an additional solar array ought to be added, and the final one closed out a number of requirements changes just prior to launch. Before seeing the network, when the solar-panel change was mentioned, one of the engineers on the project identified this first change to have been a mistake in hindsight, having caused a significant amount of issues across the spacecraft, for little actual benefit once the power needs were better understood. This goes to show that this large change cluster identified a complex issue that was clearly evident to the engineers after the fact, and anecdotally demonstrates the importance of thorough systems architecture to reduce the ambiguity as early as possible.

3.5 Agents of Change

The ECRs and affected documents from the SMAP project were handled by 59 individuals who worked on the project. Of these 59 people, 6 of them had half of the total betweenness centrality, showing that about 10% of the individuals formed the bulk of the connections for the entire group, as can be seen in figure 3-9. This correlates to a common principle seen in, for example, the Roman military hierarchical organization,
where each individual would only command 10 others, each of whom would command another 10, and so on.

Figure 3-10 shows the social network for the SMAP project, colored by the natural clustering, determined by the Louvain modularity algorithm[9]. For visualization purposes, the size of the node reflects the number of connections to the node, or its nodal degree, and the placement of nodes is determined by a force-directed algorithm that attempts to minimize the ‘energy’ described by the distance between connected nodes. Additionally, the thickness of the lines is the weight of that pairwise edge, reflecting how often that pair worked on a change together.

Two important observations are made from these graphs for SMAP. Firstly, the natural clusters match the organizational chart fairly well. The light blue cluster in figure 3-10 is made up of C&DH and ATLO engineers, and the violet cluster in the same plot are all Spacecraft/Power Systems engineers. Furthermore, the natural cluster in dark blue near the bottom of the figure is largely the Instrument team grouped together in the modularity as well.

Secondly, these plots show that system engineers and V&V engineers are strong links between clusters. For example, Sc49 is naturally clustered by modularity with the Instrument team, even though she’s on the Spacecraft team; but upon closer
Figure 3-10: Social Network for the SMAP project as reconstructed from the ECRs inspection we find that Sc49 is a V&V engineer, and thus she is providing a vital interface between the instrument design, and its integration into the spacecraft.
Chapter 4

Mars Science Laboratory

4.1 Overview of the Space Mission

The Mars Science Laboratory (MSL), now known as the Curiosity Rover and shown in figure 4-1, is a Mars rover, with eight primary mission objectives and eleven primary instruments including cameras, spectrometers, radiation detectors, environmental sensors and atmospheric sensors[10]. There are four science goals, Biological, Geological, Planetary and Surface Radiation, which are comprised of the following eight objectives[11]:

- Determine the nature and inventory of organic carbon compounds
- Inventory the chemical building blocks of life (carbon, hydrogen, nitrogen, oxygen, phosphorous, and sulfur)
- Identify features that may represent the effects of biological processes
- Investigate the chemical, isotopic, and mineralogical composition of the martian surface and near-surface geological materials
- Interpret the processes that have formed and modified rocks and soils
- Assess long-timescale (i.e., 4-billion-year) atmospheric evolution processes

Figure 4-1: Mars Science Laboratory, Courtesy NASA
• Determine present state, distribution, and cycling of water and carbon dioxide
• Characterize the broad spectrum of surface radiation, including galactic cosmic radiation, solar proton events, and secondary neutrons

MSL was launched late November 2011 and landed on Mars in August the following year, after just over eight years of development beginning with the start of Phase A in October 2003. MSL has a more complex mission design than SMAP. Not only does it have far more scientific objectives and instrumentation, but MSL also went through a transit orbit to Mars, a Mars orbit entry maneuver, atmospheric descent, and finally landing via a propulsive "sky crane" maneuver. After landing MSL was finally ready to begin the scientific mission

Table 4.1: Document and ECR counts for MSL

<table>
<thead>
<tr>
<th>MSL</th>
<th>Docs.</th>
<th>ECR</th>
<th>ECR/Doc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>307</td>
<td>657</td>
<td>2.14</td>
</tr>
<tr>
<td>Engineering Design</td>
<td>55</td>
<td>142</td>
<td>2.58</td>
</tr>
<tr>
<td>Plans or Procedures</td>
<td>1711</td>
<td>39</td>
<td>0.02</td>
</tr>
<tr>
<td>Operations</td>
<td>43</td>
<td>8</td>
<td>0.19</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>2442</strong></td>
<td><strong>940</strong></td>
<td><strong>0.38</strong></td>
</tr>
</tbody>
</table>

1 Documents and ECRs of Unspecified type are not explicitly shown.
2 Some ECRs affect more than one document type.

4.2 Timing of Change

Figure 4-2 shows the timing of change analysis for the MSL project. During MSL, as with SMAP, the document generation scaled up to a plateau following PDR, however, in this project, the path towards SIR shows a run-away behavior, exhibiting an avalanche pattern. Originally MSL was set to launch in 2009, but it was determined that the project could not meet that date and the launch was slipped two years to 2011. Figure 4-3 shows the documents and ECRs plotted cumulatively.

MSL’s document generation builds through Phase B to about 25 documents per 60 days, then remains fairly constant through CDR. Prior to SIR there is a spurt in
document generation to a maximum of about 45 documents per 60 days, at which point the schedule is re-baselined for a two year launch slip. Note that the document generation rate in this first part of MSL is lower than for SMAP, a smaller project. Once MSL is restarted, document growth accelerates rapidly to 60 documents per 60 days only tapering off just prior to launch. The ECRs show roughly the same profile. On average MSL had 0.69 ECRs per document, similar to SMAP.

MSL demonstrates a curious acceleration of change after SIR3 leading towards launch. This behavior could indicate an incomplete SIR, however, it is more likely that this represents a schedule-constrained push to wrap up changes before the launch window, since missing the launch window for a Mars mission results is a 26 month wait for the next opportunity.

For requirements artifacts, where the most change is documented, ECRs outnumber documents by more than 3:1 for MSL and the Requirements ECRs continued well into Phase D, as shown in figure 4-4. Similar to SMAP, this shows that many requirements were being changed instead of waived when not met, though in MSL,
the ECRs for requirements documents are still accelerating by launch.

Note that this ECR count is based on the number of times a Document of each type is changed by an ECR, such that a single ECR can count towards multiple types, or even multiple counts of the same type. As such, the total numbers differ from table 4.1, but should properly reflect how often these documents are being changed.

The re-baselining event moved the launch target back two years, and reset the project once it was clear that the original launch window in 2009 would be missed. The official reason for the delay is cited to be technical issues with the cryogenic actuators[12], but the change propagation data implies that there may be something deeper to missed launch window. The sharply increasing change activity between CDR and SIR2 is consistent with the avalanche profile proposed by Eckert et al.[3], and the re-baselining activity is consistent with the suggested 'drastic action' that would be necessary to end the cascading change. While this single case is not enough to sufficiently prove the behavior, it implies that the particular technical difficulties were only a portion of the cause for re-baselining.
Figure 4-4: Cumulative Artifact Releases by Document Type

(a) Requirement Type Artifacts

(b) Engineering Type Artifacts

(c) Planning Type Artifacts

(d) Operations Type Artifacts
4.3 Propagation of Change

The Propagation of Change data for MSL is presented in figure 4-5, which looks at the process layer through the propagation-by-location dimensions.

As with SMAP, the largest sources of self-directed change are the system engineering groups for MSL, having initiated over 400 ECRs that do not affect other subsystems, which is significantly more than the second largest self-directing subsystem. They also initiated the most ECRs that affect other systems and were the largest absorber of change as well (Figure 5.). This confirms that system engineers are a locus of change, as suggested by SMAP.

For MSL, the top two generators of change after systems engineering, are System Integration and Test activities and the Payload. The payloads set interface requirements on the spacecraft system while the integration and test activity uncovers issues at the system level that requires modifications.

The top absorbers of change (again, after systems engineering) were the Flight System (FS) Mechanical System, FS Avionics and Flight Software. While FS Mechanical Systems is less inherently flexible than the software subsystems, it absorbs much change from the mechanical payload subsystems. Since the FS bus is less mission critical than the science payloads, change often gets shifted to that subsystem in order to avoid inherently more risky areas.
Figure 4-5: MSL Change Propagation by Subsystem

(a) Self-Directed Change by Subsystem

(b) Generated Change by Subsystem

(c) Absorbed Change by Subsystem

71
4.3.1 ECR Parent-Child Networks

It is also possible to analyze the propagation-by-time dimension, which is presented in figure 4-6.

The majority of ECRs in the MSL project were not part of inter-ECR networks (only 269 out of 940 were connected in these networks,) which is consistent with the finding by Pasqual and deWeck\[1\], as well as the SMAP project.

The MSL ECRs that were part of parent-child clusters are shown in figure 4-6,
where it can be seen that the majority of the clusters are simple linear or branching chain, with a couple smaller tangled clusters. However, there are a few larger more complicated clusters, the three largest of which are labeled, C1, C2 and C3.

Cluster C1, shown in figure 4-7 is made up of two sub-clusters based on instrument changes to SAM (Sample Analysis at Mars, containing chemical analysis equipment), highlighted in violet, and ChemCam (Chemistry & Camera, a combined laser vaporizer and spectrograph), highlighted in red. These original clusters arose independently, largely based on thermal considerations for each instrument. The sub-cluster in cyan is a network of changes to the thermal Functional Design Documents (FDDs), that ended up linking the two instrument change clusters together. This is interesting behavior, since it reveals a functional connection between the instruments that is not part of a formal link, showing a lack of modularity and inherent complexity in the architecture.
Cluster C3, shown in figure 4-8 is more in line with the expected ECR Parent-Child behavior. This cluster is made up of many related changes in the power subsystem across the spacecraft. The upper section of this cluster, highlighted in violet is a large sub-cluster of changes based on risk assessment of the power system, the lower section is comprised of related power system changes such as the Survival Heater, and Power Switch. These two sub-clusters are related through FDD cleanups and other dependent changes, such as adding in additional power inhibitors to reduce radio frequency interferences during launch.

Interestingly, clusters C1 and C3 were originally connected in the dataset through ECR 111004, which is an update to an ICD. ICD updates generated a lot of erroneous connections, linking ECRs by time instead of parent-child relationships, since these ECRs merely updated a global document with changes from all the (unrelated) ECRs.
since the last update. As such, the connections from 111004 (and other ICD updates) were removed, and the C1 and C2 clusters were separated.

Cluster C2, shown in figure 4-9, contains a large chain of electronics software and hardware changes. The changes in the central sub-cluster, highlighted in green, are all linked to multiple issues with the FPGA. The problems with the FPGA were well known during the project, and were referred to as 'know idiosyncrasies of the as built hardware' in ECR 109498. The linear branches off this sub-cluster were all software changes that fixed various other hardware issues, and were linked with systems engineering roles, such as in ECR 111044, where V&V engineers attempted to wrap up many open issues that had been encountered through Problem Failure Reports (PFR) or identified during the risk assessment.
4.4 Agents of Change

The MSL project had 138 people involved in the ECR/document process, but only 4 of them had half of the betweenness centrality (as can be seen in figure 4-10)—which could be showing a more concentrated change process, controlled by few individuals. However, the distribution has a 'fat tail' where many more engineers have a small amount of the centrality in the network, which clearly describes a more distributed network. As such, the highly central individuals necessarily don’t control the change process, but most likely play significant connecting roles. This can be seen visually by inspecting the social graph in figure 4-11.

Figure 4-10: Betweenness Centrality and Weighted Degree of the Engineers that Contributed to MSL

As predicted by the connectedness chart, the social network is largely very distributed, with the several highly central individuals connecting various groups. For example, Pse70 in the blue cluster is largely the reason that the blue cluster is connected to the rest of the network. The placement of nodes is determined by a force-directed algorithm that attempts to minimize the 'energy' described by the distance between connected nodes.

As a reminder, in figure 4-11, for visualization purposes the size of the node reflects the number of connections to the node, or its nodal degree, and each natural
Figure 4-11: Social Network for the MSL project as reconstructed from the ECRs cluster—as determined by the Louvain modularity algorithm— is highlighted by color. Additionally, the thickness of the lines is the weight of that pairwise edge, reflecting how often that pair worked on a change together.

In the case of MSL, natural clustering does not cleanly separate the network. Each of the clusters contain individuals from many different roles, there are a significant amount of inter-cluster links, and the clusters are often locationally split by one another.

Figures 4-12 and 4-13 show the social network split up by the re-baselining date.
While the social network before re-baselining still exhibits the clumping and inter-cluster connections, the social network is much cleaner after the re-baselining date. In both networks, Sc89 and Sc85 are highly central, close to each other, and strongly connected to Pse45. Sc112 plays a much smaller role in the agents of change after re-baselining, as does Pse70, and Pse51 is less central, but still prominent. Perhaps most telling, is that before re-baselining there is a highly inter-connected knot in the area of Sc85, which is not present afterwards.
Chapter 5

Comparison of SMAP and MSL

5.1 Comparison of Timing of Change

In both SMAP and MSL, as can be seen in table 5.1, the ECRs for Requirement type documents outnumbered these documents, by almost 2:1. This shows that many requirements were being changed instead of waived when not met, and shows that in both projects the systems engineers were using the low-level requirements to re-allocate and balance margins.

Figure 5-1 shows the histogram and cumulative timing of change for SMAP and MSL side-by-side. The shape of the time profile differs significantly between SMAP and MSL. SMAP exhibits a blossom profile, demonstrating a healthy and expected change load, and the cumulative ECRs leveling off at launch showing that the change is wrapping up. MSL, on the other hand, shows two separate avalanche profiles, split by the re-baselining effort. The change accelerates towards SIR instead of plateauing in between PDR and CDR, then both ECRs and documents drop off during re-baselining, however, another avalanche is seen from re-baselining on to launch. The

<table>
<thead>
<tr>
<th>Project</th>
<th>Req Docs</th>
<th>ECRs</th>
<th>ENGR Docs</th>
<th>ECRs</th>
<th>Plan Docs</th>
<th>ECRs</th>
<th>Ops Docs</th>
<th>ECRs</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMAP</td>
<td>195</td>
<td>412</td>
<td>102</td>
<td>217</td>
<td>654</td>
<td>32</td>
<td>26</td>
<td>123</td>
</tr>
<tr>
<td>MSL</td>
<td>280</td>
<td>1001</td>
<td>38</td>
<td>153</td>
<td>1337</td>
<td>58</td>
<td>36</td>
<td>9</td>
</tr>
</tbody>
</table>

Table 5.1: Documents and ECRs of each type for SMAP and MSL
cumulative chart for MSL shows clearly that ECRs are still accelerating towards launch. Eckert et al.[3] suggest that when a change avalanche occurs change continues to rise as the design progresses until drastic action is taken. However, MSL had a successful launch during the second launch window, which suggests that this second
avalanche was an artifact of the build up of work to finish the design before launch. It is also important to note that software design for the rover phase of the mission was always planned to continue up until landing, and thus some of the change leading up to launch was expected software design work. However, the appearance and behavior of the first avalanche is consistent with Eckert et al.[3], with the ‘drastic action’ in this case being the re-baselining effort.

5.2 Comparison of Propagation of Change

SMAP and MSL exhibit many similarities with regards to the propagation of change from one subsystem to another. Some consistently strong absorbers and generators arise between these two disparate missions, the knowledge of which can have large impacts for future planning. Note that SMAP and MSL named the subsystems differently, for example the level three Flight Systems (FS) in MSL is called Spacecraft in SMAP. Since some of the level four subsystems have the same name, most subsystems are labeled with the level three as well.

5.2.1 Consistently Strong Absorbers

<table>
<thead>
<tr>
<th>SMAP Subsystem</th>
<th>Changes</th>
<th>MSL Subsystem</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spacecraft/FSSE</td>
<td>49</td>
<td>Project System Engineering</td>
<td>174</td>
</tr>
<tr>
<td>Spacecraft/C&amp;DH</td>
<td>43</td>
<td>System Integration and Test</td>
<td>154</td>
</tr>
<tr>
<td>Spacecraft/ATLO</td>
<td>37</td>
<td>Payloads</td>
<td>70</td>
</tr>
<tr>
<td>Instrument/MTS</td>
<td>33</td>
<td>FS Avionics</td>
<td>50</td>
</tr>
<tr>
<td>Spacecraft/Structure</td>
<td>25</td>
<td>Flight Software</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FS Mechanical System</td>
<td>35</td>
</tr>
</tbody>
</table>

In both projects, as can be seen in table 5.2, software is a primary absorber, with SMAP’s Command and Data Handling subsystem (Spacecraft/C&DH) and MSL’s Avionics and Flight Software subsystems. Additionally system engineering is a primary absorber in both projects; Spacecraft/FSSE and Spacecraft/ATLO from SMAP,
and Project System Engineering and System Integration and Test from MSL. Both of these effects were predicted, but surprisingly, the structural subsystems (Instrument Mechanical, Thermal & Structural (MTS), and the FS Mechanical Subsystem) were significant absorbers. This is unexpected, since hardware components are more difficult to change than documents or software, and these subsystems form the foundation of the spacecraft. However, after gathering comments from engineers who worked on the projects, this was seen as an understood behavior; when issues were identified in the spacecraft bus components or the scientific instrumentation itself, it was often less impactful to alter the supporting structures than these mission critical components when possible.

5.2.2 Consistently Strong Generators

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMAP</td>
<td></td>
</tr>
<tr>
<td>Spacecraft/FSSE</td>
<td>60</td>
</tr>
<tr>
<td>Instrument/Antenna</td>
<td>31</td>
</tr>
<tr>
<td>Project SE</td>
<td>13</td>
</tr>
<tr>
<td>Spacecraft/ATLO</td>
<td>33</td>
</tr>
<tr>
<td>Instrument/MTS</td>
<td>9</td>
</tr>
<tr>
<td>MSL</td>
<td></td>
</tr>
<tr>
<td>Project System Engineering</td>
<td>159</td>
</tr>
<tr>
<td>FS Avionics</td>
<td>52</td>
</tr>
<tr>
<td>Payloads</td>
<td>38</td>
</tr>
<tr>
<td>FS System Engineering</td>
<td>23</td>
</tr>
<tr>
<td>System Entegration and Test</td>
<td>19</td>
</tr>
</tbody>
</table>

As expected, Systems Engineering is a large generator of change in both projects, as shown in table 5.3. Similarly expected is that the scientific payloads were significant generators of change as well. One of the other primary absorbers for each project also turned out to be primary generators; Instrument/MTS from SMAP and FS Avionics from MSL. While this could be due to these subsystems having a certain likelihood of generating change propagation on their own, it could also be the case that these subsystems are passing on some of the incoming change in the form of generated changes for other subsystems, rather than closing all of the change propagation chains.
5.2.3 Change Propagation Index

The Change Propagation Index (CPI) was introduced to post-hoc change propagation analysis by Giffin et al.[4], extending the concept originating from Suh et al.[7] shown in equation 1.1. In order to clarify the calculation, Giffin et al. proposed equation 5.1 to calculate the CPI of element $i$.

$$CPI(i) = \frac{C_{out}(i) - C_{in}(i)}{C_{out}(i) + C_{in}(i)}$$ (5.1)

This formulation was used to calculate the CPI of the subsystems for both projects, the results of which are shown in figure 5-2 and listed in Appendix B. The CPI is limited by the fact that it is a one-dimensional metric, that is normalized to the amount of incoming and outgoing change for each subsystem. As such, it expresses limited information, for example, a subsystem that is only involved in a single outwardly-directed change will show an extreme CPI of -1 or 1, implying that it is a strong absorber or generator, respectively, when it might otherwise be considered a constant.

However, the CPI does present a simple method to quickly compare the subsystems and projects through a quantitative measurement. The CPI shows similar characteristics as sections 5.2.1 and 5.2.2. For example, software-based subsystems such as Spacecraft/GNC in SMAP and Flight Software and FS GNC in MSL have negative CPIs in the range of -0.4 to -0.6, showing that they absorb more change than they generate, and Project Science and Sample Acquisition in MSL, and Instrument/ Antenna and Instrument/Radar in SMAP have positive CPIs, indicating that they generate more change that they absorb, which is consistent with section 5.2.2.

5.2.4 Implications for future missions

Knowing where to expect change to occur, and in particular propagate to and from, could be very helpful when planning a project. Primarily, resource and staffing plans can be adjusted to account for the relative change frequency in addition to the planned engineering work.

Additionally, propagation estimates can be used when initially setting margins, if
a subsystem is expected to need to absorb a lot of change for other areas, allocating larger initial margins would allow for smoother design changes when this propagation occurs. This could be particularly valuable for future projects regarding the Instrument/MTS and FS Avionics since there is a likelihood that these subsystems show up as primary absorbers and generators because they could not absorb all of the change directed towards them and had to propagate some of the change to other subsystems. If these subsystems had larger built-in margins, they may have been able to be final absorbers for more of the propagation.

Finally, this data can be used to inform design for flexibility in future projects. While designing in flexibility at the beginning of a project can add to the initial workload, it will simplify the re-design process, allowing designs to easily adapt to incoming change. This analysis has shown that the payload supporting Mechanical/Termal/Structures subsystem would benefit from design for flexibility, since both SMAP and MSL saw a significant amount of incoming change in this subsystem. Thus, the initial investment in design for flexibility will likely provide a net benefit, since these subsystems will probably require much re-design effort.
5.3 Comparison of Agents of Change

The geometry of the SMAP (figure 3-10) and MSL (figure 4-11) networks is significantly different. SMAP’s natural clustering separates into four roughly equally sized groups with one smaller one, while MSL separates into five similarly sized groups. However, MSL’s groups could not be disentangled in two dimensions (the yellow group crosses over the other groups, and the purple group separates the green) while SMAP’s groups are separable.

Since the positioning of the nodes is based on reducing the free energy of the edges, and the clustering is based on reducing the number of inter-cluster edges, the fact that MSL’s groups are tangled suggests that this network is highly distributed, with many light connections spanning the network, and very little hierarchical structure.

This is confirmed when looking at the organization of each group. Each of SMAP’s groups has at least one node that is significantly larger than the rest (i.e., has a high betweenness centrality,) indicating that he or she is a central point of contact. MSL’s larger nodes are concentrated in the red group, with one in the blue group and none in the three other groups. Hence, the ECR process for the whole MSL system is directed largely from the red group. Also, for MSL there are many one-on-one interactions and significant number of connections across groups. For both projects, however, the larger nodes are system engineers or system integration and test engineers who serve as the ECR communicators.

The content of each of the groups was also examined in more detail. For SMAP, each group contains a single subsystem (such as avionics, instruments, structure, or the mission system) combined with either system engineers and/or system integration and test engineers. Additionally, the structure group, in violet, is at the center of the other groups; consistent with the earlier observation that the structure can serve as an absorber of change.

MSL exhibits very different behavior; the groups do not naturally organize by a single subsystem. Each group has representatives of all the disciplines, perhaps showing a system of systems architecture in contrast to SMAP’s single system archi-
tecture. The disciplines in each group are much more mixed, and most disciplines are present in all groups. The purple group for payload is the most similar to SMAP’s, however payload is represented in other groups as well.
Chapter 6

Manifestation of Complexity

The analyses so far have been straight-forward multi-layer, multi-dimensional analyses of the design process and change propagation. While there are many insights to be gained from these tools and metrics, they all consider the direct impacts of the data. In order to further investigate complexity the following two tools go beyond the surface of the data to look at how complexity is manifested throughout the engineering change process.

6.1 Number of Subsystems Affected per ECR

One way to look at the complexity is related to the number of ECRs that affect multiple subsystems, when the relationship is identified at the time the ECR is initiated. Table 6.1 lists this metric for SMAP and MSL.

Table 6.1: Comparison of the Number of ECRs that affect Multiple Subsystems

<table>
<thead>
<tr>
<th># Subsystems Affected</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMAP</td>
<td># ECRs</td>
<td>507</td>
<td>45</td>
<td>13</td>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>% of ECRs</td>
<td>89%</td>
<td>8.8%</td>
<td>0.8%</td>
<td>2.5%</td>
<td>0.2%</td>
<td>0</td>
</tr>
<tr>
<td>MSL</td>
<td># ECRs</td>
<td>755</td>
<td>121</td>
<td>47</td>
<td>11</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>% of ECRs</td>
<td>80%</td>
<td>12.8%</td>
<td>5%</td>
<td>1.2%</td>
<td>0.2%</td>
<td>0.2%</td>
</tr>
</tbody>
</table>

Most ECRs affected only 1 subsystem, 507 out of 570 for SMAP (89%) and 755 out of 940 for MSL (80%). However, there are a few that affect multiple subsystems,
MSL has a higher percentage of ECRs that affect multiple subsystems than SMAP, and also has ECRs that affect a greater number different subsystems than the ECRs in SMAP. MSL has two ECRs affecting seven different subsystems, compared to SMAP which has a single ECR that reaches five subsystems its peak. This reflects MSL’s more complex character via this intra-ECR metric.

6.2 ECR Parent Child Relationships

A second way to investigate complexity is related to the number of ECRs that spawn additional ECRs in "parent-child" clusters, creating inter-ECR links. As opposed to the intra-ECR metric described in section 6.1, parent-child relationships are created when there is a delay in the realization that the change has propagated, resulting in another ECR initiated later in time. These parent-child relationships are mapped into a network, shown in figure 3-8 for SMAP and figure 4-6 for MSL, and the cluster sizes are recorded in table 6.2.

These relationships reveal a different level of complexity than direct ECR-subsystem relationships described by the intra-ECR links, since (some of) the effects of the parent ECR were not considered during its tenure and thus a separate ECR needed to
be created at a later date. This distinction highlights the human aspects of dealing with complex systems, and reveals the apparent complexity of the system.

Since MSL is a larger project with more ECRs, it would be expected that it would generate more ECR Parent-Child (P-C) networks, however, this is not enough to account for the number and size of ECR P-C clusters in MSL. While SMAP has only one cluster larger than 8 ECRs, MSL has 6, with three clusters larger than 20.

Interestingly, the closer inspection of these clusters reveals different characteristics then the large cluster in SMAP. As discussed in the ECR P-C analysis in section 4.3, C3 is comprised of changes relating to the power structure which serves as an important interface between subsystems. This Cluster is similar to the large cluster in SMAP, which is a slew of changes over the duration of the project, also connected by the power structure. However, the C1 cluster in MSL shows a different characteristic. This cluster is made up of a family of changes from each of two different scientific instruments, connected by a cluster of thermal design changes. While this also shows change propagating through subsystems via the interfaces, C1 shows localized change networks only connecting after the fact through the interface clean up.

The inter-ECR tool of Parent-Child relationships has revealed the distribution of ECR P-C clusters, as well as the more varied nature of these clusters during MSL, and shows that MSL has a more complex character in this regard as well.
6.3 Other Potential Complexity Metrics

There has been some research into more tools and metrics for quantitatively analyzing complexity, the most promising of which is the result of extending the concept of Brownian motion to complex systems design.

6.3.1 Sinha’s Equation

Kaushik Sinha\cite{13} proposes a metric that measures the "structural complexity of an engineered complex system, equation 6.1, motivated by the Hamiltonian energy of organic molecular systems.

\[
Structural \ Complexity, \ C = C_1 + C_2C_3 \quad (6.1)
\]

This metric has two terms, the first of which considers the components alone, and is the summation of the individual complexities of the elements, \( C_1 \). The second term considers the architecture of the system, and is comprised of the number and complexities of the links between elements, and the architectural complexity in the arrangement of elements and links. \( C_1, C_2 \) and \( C_3 \) can be expanded, resulting in equation 6.2, where \( \alpha \) is the element complexity, \( \beta \) is the link complexity, and \( E(A) \) is the graph energy of the adjacency matrix \( A \).

\[
C = \sum_{i=1}^{n} \alpha_i + \left( \sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{ij}A_{ij} \right) \frac{E(A)}{n} \quad (6.2)
\]

While the graph energy is well understood, and the link complexity can often be, at a minimum, rank ordered by link types, the difficulty in applying this metric lies in determining the element complexities. Even in the case of spacecraft with similar subsystems, the individual subsystems of each unique spacecraft are dissimilar enough to preclude consistent estimation. It may be possible to recursively apply this metric down to the component level, set comparative component complexities, and then rebuild the top level complexity. However, this requires a single exhaustive catalog
of the spacecraft design containing not just every component, but every link and interface between them. Since this data is not available for these space missions, the complexity can not be calculated according to Sinha’s equation.

6.3.2 Sinha’s Equation applied to Engineering Change

Sinha’s equation could potentially be extended to the heuristic subsystem propagation data. While this would not be able to calculate the basic structural complexity of the final product, it would reveal the apparent complexity of the design process, with aspects of both the product complexity, and organizational and procedural complexity. Thus, the "process complexity" can be defined in a similar manner as Sinha’s structural complexity.

I propose that this process complexity can be calculated from the subsystem propagation network which is constructed from the change propagation data in sections 3.4 and 4.3. The links between subsystem are directed connections created from outwardly-directed changes, with the source node as the generator, and the target node as the absorber of the change, and weighted by the number of ECRs with that connection. These edges can fulfill the $C_2$ and $C_3$ terms in the structural complexity equation (6.1), with $C_2$, the link complexity, defined as the weight of each edge multiplied by it’s directedness (1 for a directed edge, and 2 for a bi-directed edge), and $C_3$ similarly defined as the Graph Energy of the network divided by the number of elements.

Since the process complexity uses the change process as the indicator of complexity, the element complexity can be calculated form the heuristic change data. As such, the element complexity, $C_1$, can be defined as the internal change of a subsystem during the project, and is thus the magnitude of self-directed change for each subsystem. With this final piece, the process complexity equation can be aggregated, and takes the form of equation 6.3.

\[
Process \ Complexity, \ C_p = \sum_{i=1}^{n} \mu_i + \left( \sum_{i=1}^{n} \sum_{j=1}^{n} \phi_{ij}A_{ij} \right) \frac{E(A)}{n} \quad (6.3)
\]
where:

\[ \mu = \text{Number of Self-Directed ECRs of the } i^{th} \text{ subsystem} \]

\[ \phi = \text{Number of Edge directions between subsystems } i \text{ and } j \text{ (1 or 2)} \]

\[ A = \text{Weighted Adjacency Matrix} \]

\[ E(x) = \text{Graph Energy} \]

\[ n = \text{number of Subsystems} \]

Considering this definition, the process complexity of both the SMAP and MSL projects can be computed from the subsystem propagation data recorded in appendix A. Table 6.3 shows the results of this calculation, and as postulated, MSL has a higher complexity than SMAP based on this metric. However, more projects will need to be analyzed using this metric in order to establish statistical rigor behind it.

Table 6.3: Process Complexity Results

<table>
<thead>
<tr>
<th>Project</th>
<th>Process Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMAP</td>
<td>367</td>
</tr>
<tr>
<td>MSL</td>
<td>492</td>
</tr>
</tbody>
</table>
Chapter 7

Conclusions

The timing of change behavior was very different between SMAP and MSL, the former following a healthy blossom profile, and the later exhibiting two separate avalanche profiles before reaching a successful launch after re-baselining and restarting the project after the first avalanche. While it is likely that the second avalanche towards launch is an artifact of a compressing design schedule to reach the launch date, the first avalanche exhibits all of the defining behaviors of an avalanche according to Eckert et al., including the drastic action that ended it.

The agents of change were also quite different between SMAP and MSL. The agents of change analysis showed that SMAP followed a hierarchical, and relatively discipline-based organization of the contributors. Across the entire project, MSL showed a distributed and tangled organization, as it did when only looking at the project before re-baselining. However, when isolating the project after the re-baselining, the social network followed a much more organized configuration, which may have been a contributing factor in the projects ultimate success.

In both of the projects, system engineering roles acted as significant connections throughout the social networks.

SMAP and MSL both had very similar propagation behavior with regards to the most common generators and absorbers of change. The understanding of these common behaviors can be used to inform project planning for space missions in the future. Ideally, more projects would be analyzed in a similar way to build statistical
significance into these trends, and potentially identify differences in mission type such as earth-orbiters, planetary rovers or deep-space probes.

Furthermore, the two-dimensional analysis from Eckert et al. [3], shown in figure 1-4, could be filled out once this data has been correlated. The borders between Carriers and Buffers, and Absorbers or Generators are only qualitatively defined, and it may be interesting to quantitatively investigate these borders once that mapping is complete.

One way to look at the apparent complexity of design projects is to investigate the immediate and latent change propagation metrics. Both of these metrics can reveal much about the character of the complexities of these projects. In both cases, MSL exhibited a more complex character, outclassing SMAP in both the number of subsystems affected by single ECRs, and the number and complexity of ECR parent-child networks.

The Hamiltonian energy formulations for Brownian motion can be extended to complexity measurements, as postulated by Sinah. While the rigorous computation thereof requires absolute, detailed information about the complete system, this equation can be extended to look at the past change data. Since the change data is convoluted by the change process and the individuals executing it, this complexity metric cannot evaluate the absolute complexity of the product. However, this could be an effective metric of apparent complexity.

7.1 Recommendations for Future Work

While this research set up a thorough multi-layer and multi-dimensional analysis of change propagation, there remain important areas for future work:

- It could be highly valuable to use this methodology to analyze multiple projects, which would present the opportunity to establish statistical data behind some of the systematic trends that were identified.

- It may also be useful to analyze projects across multiple classes of space mis-
sions, comparing and contrasting between earth orbiters, planetary rovers and deep space probes. It is postulated that most of the change propagation behaviors would be common across these classes, but that some systematic differences would emerge.

- While the one-dimensional Change Propagation Index was computed for the subsystems in both the SMAP and MSL projects, the two dimensional equivalent of Propagation Behavior, shown in figure 1-4, could elucidate some qualities of the subsystem propagation more clearly. This would be particularly interesting when combined with a correlated subsystem data set from multiple projects. The overall layout of the propagation behavior tool is merely proposed in Eckert et al.[3], and a large data set could be used to establish the locations of the different behavior regions more precisely.
Bibliography


Appendix A

Subsystem Propagation Data
Table A.1: SMAP Subsystem Propagation Adjacency Matrix

|-----------------|----------------------|------------------|-------------------|----------------|---------------|----------------|-------------------|-------------------------|----------------|-----------|----------------|----------------|----------------|-----------------|----------------|----------------|----------------|-------------------|------------------|-------------------|----------------|---------------- |
Table A.2: MSL Subsystem Propagation Adjacency Matrix

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FS Avionics</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS GNC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS Mechanical System</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS Power</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS Propulsion</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS System Engineering</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS Telecom</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FS Thermal Control</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Flight Avionics</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Flight Software</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Ground Data System Mgmt.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mission Assurance</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mission Design</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mission Ops. System</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Payloads</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Planetary Protection</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Project Mgmt.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Project Science</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Project System Eng.</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Simple Acquisition</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>System I&amp;T</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Unknown</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Appendix B

Change Propagation Indexes

Tables B.1 and B.2 contain the full lists of the Change Propagation Indexes for all subsystems of SMAP and MSL respectively. Subsystems without a CPI number did not have any absorption or generation of change, and thus the CPI is undefined for these subsystems.
Table B.1: SMAP Change Propagation Indexes

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>CPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instrument / I&amp;T</td>
<td>-1</td>
</tr>
<tr>
<td>Instrument / Radiometer</td>
<td>-1</td>
</tr>
<tr>
<td>Spacecraft / Propulsion</td>
<td>-1</td>
</tr>
<tr>
<td>Instrument / SE</td>
<td>-0.83</td>
</tr>
<tr>
<td>Spacecraft / Structure</td>
<td>-0.72</td>
</tr>
<tr>
<td>Mission System / MOC</td>
<td>-0.69</td>
</tr>
<tr>
<td>Spacecraft / C&amp;DH</td>
<td>-0.65</td>
</tr>
<tr>
<td>Instrument / MTS</td>
<td>-0.57</td>
</tr>
<tr>
<td>Mission Assurance</td>
<td>-0.55</td>
</tr>
<tr>
<td>Spacecraft / ATLO</td>
<td>-0.49</td>
</tr>
<tr>
<td>Spacecraft / I&amp;T</td>
<td>-0.2</td>
</tr>
<tr>
<td>Spacecraft / Telecom</td>
<td>0</td>
</tr>
<tr>
<td>Spacecraft / FSSE</td>
<td>0.10</td>
</tr>
<tr>
<td>Mission System / SOC</td>
<td>0.14</td>
</tr>
<tr>
<td>Spacecraft / Power</td>
<td>0.14</td>
</tr>
<tr>
<td>Mission System / Risk Mgmt.</td>
<td>0.2</td>
</tr>
<tr>
<td>Instrument / Radar</td>
<td>0.23</td>
</tr>
<tr>
<td>Spacecraft / Thermal</td>
<td>0.23</td>
</tr>
<tr>
<td>Spacecraft / GNC</td>
<td>0.33</td>
</tr>
<tr>
<td>Project SE</td>
<td>0.68</td>
</tr>
<tr>
<td>Instrument / Antenna</td>
<td>0.82</td>
</tr>
<tr>
<td>Business Office</td>
<td></td>
</tr>
<tr>
<td>Launch Vehicle</td>
<td></td>
</tr>
<tr>
<td>Mission System / SE</td>
<td></td>
</tr>
<tr>
<td>Project Mgmt.</td>
<td></td>
</tr>
<tr>
<td>Spacecraft / Mgmt.</td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
</tr>
</tbody>
</table>
Table B.2: MSL Change Propagation Indexes

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>CPI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown</td>
<td>-1</td>
</tr>
<tr>
<td>FS Telecom</td>
<td>-0.81</td>
</tr>
<tr>
<td>Mission Operations System</td>
<td>-0.79</td>
</tr>
<tr>
<td>System Integration and Test</td>
<td>-0.78</td>
</tr>
<tr>
<td>FS Thermal Control</td>
<td>-0.53</td>
</tr>
<tr>
<td>Flight Software</td>
<td>-0.48</td>
</tr>
<tr>
<td>FS GNC</td>
<td>-0.43</td>
</tr>
<tr>
<td>FS Mechanical System</td>
<td>-0.37</td>
</tr>
<tr>
<td>FS Power</td>
<td>-0.33</td>
</tr>
<tr>
<td>Payloads</td>
<td>-0.30</td>
</tr>
<tr>
<td>Mission Assurance</td>
<td>-0.11</td>
</tr>
<tr>
<td>Project System Eng.</td>
<td>-0.045</td>
</tr>
<tr>
<td>Ground Data System Mgmt.</td>
<td>0</td>
</tr>
<tr>
<td>FS Avionics</td>
<td>0.020</td>
</tr>
<tr>
<td>FS Propulsion</td>
<td>0.091</td>
</tr>
<tr>
<td>Project Science</td>
<td>0.23</td>
</tr>
<tr>
<td>Mission Design</td>
<td>0.33</td>
</tr>
<tr>
<td>FS System Engineering</td>
<td>0.70</td>
</tr>
<tr>
<td>Sample Acquisition</td>
<td>1</td>
</tr>
<tr>
<td>Planetary Protection</td>
<td></td>
</tr>
<tr>
<td>Project Mgmt</td>
<td></td>
</tr>
</tbody>
</table>