SOCIO-COGNITIVE ANALYSIS OF ENGINEERING SYSTEMS DESIGN: SHARED KNOWLEDGE, PROCESS, AND PRODUCT

by

Mark Sean Avnet

S.B. Physics, Massachusetts Institute of Technology, 2001
M.A. Science, Technology, and Public Policy, The George Washington University, 2005

SUBMITTED TO THE ENGINEERING SYSTEMS DIVISION IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY IN ENGINEERING SYSTEMS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2009

© 2009 Massachusetts Institute of Technology. All rights reserved.

Signature of Author ____________________
Engineering Systems Division
April 6, 2009

Certified by ____________________
Annalisa L. Weigel
Jerome C. Hunsacker Assistant Professor of Aeronautics and Astronautics and Engineering Systems
Thesis Supervisor

Certified by ____________________
Christopher L. Magee
Professor of the Practice of Mechanical Engineering and Engineering Systems

Certified by ____________________
John S. Carroll
Morris A. Adelman Professor of Management
Professor of Behavioral and Policy Sciences and Engineering Systems

Certified by ____________________
Richard P. Binzel
Professor of Planetary Sciences

Accepted by ____________________
Nancy Leveson
Professor of Aeronautics and Astronautics and Engineering Systems
Chair, Engineering Systems Division Education Committee

[This page intentionally left blank]

SOCIO-COGNITIVE ANALYSIS OF ENGINEERING SYSTEMS DESIGN: SHARED KNOWLEDGE, PROCESS, AND PRODUCT

by

Mark Sean Avnet

Submitted to the Engineering Systems Division on April 6, 2009
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Engineering Systems

ABSTRACT

This research is based on the well-known but seldom stated premise that the design of complex engineered systems is done by people – each with their own knowledge, thoughts, and views about the system being designed. To understand the implications of this social dimension, the Integrated Concurrent Engineering (ICE) environment, a real-world setting for conceptual space mission design, is examined from technical and social perspectives. An integrated analysis demonstrates a relationship among shared knowledge, process, and product.

The design process is analyzed using a parameter-based Design Structure Matrix (DSM). This model, consisting of 682 dependencies among 172 parameters, is partitioned (reordered) to reveal a tightly coupled design process. Further analysis shows that making starting assumptions about design budgets leads to a straightforward process of well-defined and sequentially-executed design iterations.

To analyze the social aspects, a network-based model of shared knowledge is proposed. By quantifying team members' common views of design drivers, a network of shared mental models is built to reveal the structure of shared knowledge at a snapshot in time. A structural comparison of pre-session and post-session networks is used to compute a metric of change in shared knowledge. Based on survey data from 12 design sessions, a correlation is found between change in shared knowledge and each of several system attributes, including technological maturity, development time, mass, and cost.

Integrated analysis of design process and shared knowledge yields three interdisciplinary insights. First, certain features of the system serve a central role both in the design process and in the development of shared knowledge. Second, change in shared knowledge is related to the design product. Finally, change in shared knowledge and team coordination (agreement between expected and reported interactions) are positively correlated.

The thesis contributes to the literature on product development, human factors engineering, and organizational and social psychology. It proposes a rigorous means of incorporating the socio-cognitive aspects of design into the practice of systems engineering. Finally, the thesis offers a set of recommendations for the formation and management of ICE design facilities and discusses the applicability of the proposed methodology to the full-scale development of complex engineered systems.

Thesis Committee:

Annalisa L. Weigel (Chair)
Assistant Professor of Aeronautics and Astronautics and Engineering Systems

Christopher L. Magee
Professor of the Practice of Mechanical Engineering and Engineering Systems

John S. Carroll
Professor of Behavioral and Policy Sciences and Engineering Systems

Richard P. Binzel
Professor of Planetary Sciences

Acknowledgments

The alternate title of this thesis is “Engineers Are People Too” because it is all about the critical role that the people play in the process. Well, this is true not only for systems engineering but also for completing a Ph.D. In fact, judging from my own experience, I would say that the people are what make it happen. My name might be the one that appears on the by-line of this dissertation, but it is really the product many people’s efforts.

First, I would like to thank my advisor, Annalisa Weigel. You have always given me just the right amount of both encouragement and criticism. When I would start to veer off course, you would point it out without hesitation, help me figure out how to get back on track, and then let me find my way again from there. And when I did do that, I knew the praise that followed was well-deserved. You treat your students as individuals and take the time to figure out how best to work with each of them. I couldn’t imagine a better choice to guide me through this journey.

I would also like to thank the rest of my committee, Chris Magee, John Carroll, and Rick Binzel. Chris, I want to thank you for your support and encouragement, your thoughtful (and delightfully creative) ideas, and your quick and complete comments on anything that I sent you. John, thank you for that push I needed early on to find my way through the mind-bogglingly large and complex literature on teams and organizations and for always making yourself available to chat about research even during difficult times. And finally, Rick, I owe you a huge debt of gratitude. When other scientists thought that my ideas were at best impractical and at worst nonsensical, you were my champion and my valuable liaison to the scientific community. Although the S.M. in Planetary Science ultimately was not part of my path, I have learned more from working with you than you probably realize.

This thesis is a combination of the efforts of so many supportive and generous people at NASA Goddard Space Flight Center (GSFC). First and foremost, I would like to extend my deep appreciation to Deborah Amato, the person that made all my other working relationships at GSFC possible. Probably more than anyone else, you have embodied to me the importance of people in the Ph.D. process. I was able to gain access to the Mission Design Lab (MDL) only because you campaigned for me and my work even before I realized it. Thank you for giving me that critical foot in the door and then for being available for an occasional Indian buffet lunch and great conversation even long after you had moved on to bigger and better things.

This research would not have been possible without the contributions of John Martin, the MDL Team Lead, a man whose boundless energy at times made me forget that he is nearly five decades my senior but whose wealth of wisdom and expertise would always remind me of it. John, I would like to thank you for all of the hard work that you put into helping me with this research. Whether it was continuously reminding the team to fill out the surveys or painstakingly examining that giant matrix, you helped to make this research what it is. You once told me that you hoped this work would help you “to be a better Team Lead.” Well, I am honored to pass on to you the fruits of this labor, and I look forward to learning how you end up using it. On the other hand, I would also like to hear soon that you are entering your well-deserved and long-overdue retirement. Congratulations on an amazing career!

In addition, I would like to thank the rest of the management staff of the Integrated Design Center (IDC). Bruce Campbell, manager of the IDC, and Tammy Brown, Team Lead of the Instrument Design Lab (IDL), have both provided invaluable support and encouragement. I want to let you both know that I will be on the lookout for new Ph.D. students to come in and extend my work to the IDL. In my mind, it is the most pressing and important piece of future work coming out of this thesis, and I sincerely hope that you find some of these MDL-based results applicable there as well. Gabe Karpati, thank you for your support and your insights during the brief time when we overlapped in the IDC. Jennifer Bracken, thank you for recently taking up Deborah's role of supporting my work even though you did not know me well.

To the IDC support staff – Felicia Buchanan-Jones, Dawn Daelemans, Elfrieda Harris, Erica Robinson, and Ed Young – thank you for all of your help in making sure that I had access to everything that I needed to complete my work. You all went above and beyond anything that I would have expected or even hoped for to help me complete this work.

And, of course, to the MDL engineers, I cannot fully express my gratitude to you for spending your own valuable time reading my incessant e-mails, filling out my surveys, and talking to me at length about who you get what information from and why. You really are the life's blood of this research. It simply would not exist if not for you. So I would like to thank you all. And I would especially like to thank those engineers that were staffed on most or all of the studies that I observed. You have contributed so much, and I am grateful not only for the effort you put forth but also for the stimulating conversations that we had. Libby Adelman, Dave Batchelor, Bob Beaman, Aron Brall, Ed Brinker, Kim Brown, Anel Flores, Doug Freesland, Kequan Luu, Mike Nemesure, Dave Olney, Dave Peters, Larry Phillips, Ivonne Rodriguez, Harvey Safren, Steve Tompkins, and Ron Vento – thank you all. This research is as much yours as it is mine.

Also, I would like to acknowledge the NASA Graduate Student Researchers Program (GSRP) for supporting this work. I would like to thank Mark Steiner for serving as my GSRP advisor and Bruce Campbell for securing IDC funds to cover the first year of support. I would like to acknowledge the MIT Program on Emerging Technologies (PoET) and the NSF Integrative Graduate Education and Research Traineeship (IGERT) program for funding the first two years of my Ph.D. I also want to thank the PoET faculty – Dan Hastings, Dava Newman, Ken Oye, and Roe Smith – whose guidance was so valuable early in my work.

I would like to express my appreciation for Joe Kirk. Joe, we've never actually met, but you gave of your own time to modify your loop-counting Matlab script to do what I needed it to do. It provided robust support for the loop analysis in Chapter 4, and Appendix C is really the product of your work, not mine. Thank you.

To all of the students that I have worked with, thank you for the help, guidance, and friendship. First, thank you to all of the CASPARites (whatever that stands for!) that I have been honored to call colleagues – David Broniatowski, Norma Campos, Thomas Coffee, Alex Cosmas, Laura Escudero, Abe Grindle, Jenn Gustetic, Masa Hashimoto, Jenny Hu, Karen Marais, Charlotte Mathieu, Greg O'Neill, Zoe Szajnfarber, Scott Van Broekhoven, and Danielle Wood. And special thanks to David and Zoe. David, you were the first person to tell me about CASPAR back when we were in Australia, and then you set up a lunch meeting with Dava Newman in

Vancouver that played a critical part in my decision to apply to ESD. Zoe, you have always been supportive of my work, and you were always eager to talk about research, statistics, the ESD Ph.D. program (the good, the bad, and the ugly!), and anything else whether it was relevant or not. I also want to thank all of my ICAT (International Center for Air Transportation) office mates – Philippe Bonnefoy, Masha Ishutkina, Marisa Jenkins, Rahul Kar, Brice Kunzi, Alex Mozdzanowska, and Roland Weibel. And lastly, I want to thank my three worthy "opponents" in this everyone-wins race to the Ph.D. – Gergana Buonova, Masha Ishutkina, and Matt Richards. It's been a fun road!

Along the way, I've had fantastic support from so many other students as well. Mo-Han Hsieh, you provided valuable insights on networks that helped me to find many of the analysis tools that I used in this research. Mike Hanowsky, thanks for the chats, the lunches, and the drinks at random times. I finally understand that intensity of the last few months of this process! João Castro, thank you for our early collaboration and for your interest in my work. Have you found any further use for that data set I gave you? Gautam Mukunda, thank you for introducing me to a diverse set of people and to a social world that I never would have expected. And, of course, to the rest of the Belize scuba quartet – Brandon Owens, Katherine Dykes, and Yves Boussemart – fist bumps all around! Brandon, I'm hot on your trail… I guess we both do *some* work every now and then, huh? Katherine, a special thanks from "the enemy" – I think we really are just a little too much alike. Yves, thank you for organizing the trip and for certifying me as an advanced diver. We really need to get our proposal together for that research on shared mental models in dive buddies. Research funding and warm tropical waters… what a combination!

And now I would like to make some special Acknowledgments to the important people in my life who gave me unending support and encouragement throughout the process.

Mom (Cindy Avnet) – Ever since I left for college, you have continued to be an amazing source of love and support. Whether it was visiting me at school, calling to check in during the hardest times, or supplementing my meager student income so I could live comfortably and work productively, I knew I could count on you. I love you, Mom.

Dad (Steven Avnet) – This is really your achievement as well as mine. You have always been there for me when I needed you, and you taught me the strength and independence that serve me so well today. Your guidance has been invaluable to me throughout this journey. I am so glad that I can live up to your 29th birthday present by getting my Ph.D. before 30! I love you, Dad.

Tracey Avnet – To my loving stepmother, I want to thank you for believing in me. You have been supportive and encouraging about anything that I've attempted to do. And you've always been willing to lend a helping hand… and a car when I'm home in Florida! You have brightened up the Avnet family, and for that, we all should be eternally grateful. Thank you!

Jasmine Avnet – My baby sister and my inspiration for the future, I am so excited about your high school graduation next year. But weren't you just that incredibly cute little five-year-old at mine? Before you know it, you'll be writing about me to mark your own achievements. So consider this your charge… I know you'll be a huge success no matter what you choose to do!

Maj. (Sel.) Adam Avnet, USAF – How can two people that are so different from each other be brothers? Well, we might be different in many ways, but we are also the same in many more. Throughout my life, I've probably spent more time with you than with any other person. Despite any disagreements we may have, we're always there for each other when it really matters.

Shiri Avnet – My wonderful sister-in-law, the newest addition to the Avnet family, and the best Israeli import since falafel, I want to thank you for everything. Staying with you and Adam was often the highlight of my countless research trips. My work won't be sending me your way anymore, but rest assured that I will find my way to crash your home again sometime soon.

Dana Brown – You've been such a caring and devoted cousin that sometimes I think you could just as easily be my big sister. You and I have seen each other through so many critical life achievements. I look forward to sharing yet another one with you at your graduation in May. I am so proud of you and thrilled that you will be there to share this accomplishment with me.

Lauren Langford – How long have we been friends again? We've been there for each other through the best and the worst. Thank you for your friendship, for understanding my conspicuous near-disappearance in the past few months, and for being there for the exciting next step in my life. I don't know if "MSA is always right," but he definitely can pick a good friend!

Matt Feinberg – I think we're way overdue for a guys' night! Even though we sometimes fall out of touch for unreasonably long times, our friendship is always rock solid, and I know that I can always count on you when I need it the most. Thank you for being there when I was preparing for my general exam. I don't remember how many times you listened to me go through my orals presentation in your living room, but I also can't tell you how much that meant and how much it helped me.

Ashley Aronow – We have known each other for almost five years, and you've been challenging my assumptions and keeping me on my toes the whole time! And only recently, you have become a really special and amazing part of my life. It all happened just as I was entering the most stressful and time-consuming part of getting a Ph.D. Through it all, you have been there for me more than I ever could have expected or asked for or even imagined. Whether it was bringing a sandwich to my office, keeping me well-caffeinated, being an incredible source of encouragement through so many long nights, or *reading and editing this entire thesis*, you have done nothing but give of yourself to make this arduous process as easy as it possibly can be. I will never forget all that you have done for me. Thank you for being there for me even through such a difficult time in your own life. I owe you more than you even realize. I love you, Ashley.

Thank you all for everything that you have done for me. I owe my Ph.D. and all of my future successes to each and every one of you. Please know that you have been and will continue to be an important part of my journey.

- Mark S. Avnet, Ph.D.

Biographical Note

Mark S. Avnet holds an S.B. in Physics with a minor in Spanish from the Massachusetts Institute of Technology and an M.A. in Science, Technology, and Public Policy from The George Washington University. Mark received a Graduate Certificate in Applied Science, Space Studies from the University of South Australia for completion of the Summer Session Program offered by the International Space University.

Prior to returning to MIT to pursue his Ph.D., Mark worked as a Program Specialist for Centennial Challenges at NASA Headquarters. He has served as a Research Assistant at The George Washington University's Space Policy Institute and as a visiting researcher at NASA Ames Research Center. In addition, he spent two years working as a software developer for Rocket Software, Inc. in Newton, MA.

In his work, Mark is driven by a desire to understand the way the world works at a fundamental level. When he did not find the answers that he sought in the equations and lab experiments of quantum mechanics, he decided that the most interesting phenomena are not determined by the behavior of tiny particles but rather by the behavior of people. Based on this realization, Mark shifted his focus from physics to policy when he entered graduate school to pursue his Master’s degree. Mark’s Ph.D. thesis represents his attempt to combine these interests and to expand his understanding of the physical and the social worlds and how they interact.

Mark’s varied research interests have led him to serve as author or co-author on journal articles and conference papers covering several space-related topics that bridge the technical and the social. These topics include the feasibility of the space elevator as a means of accessing Earth orbit, a framework for understanding and responding to threats from near-Earth objects (i.e., asteroids and comets), the role of international cooperation in achieving space science objectives, and the uses of the Moon in preparing for human exploration of Mars.

Currently, Mark’s research interests lie at the intersection of social and organizational psychology and systems engineering. For his Ph.D., he has explored this topic in the context of space mission design. In the future, he would like to apply the methods that he used in this work and related techniques to similar problems across a variety of industries and projects. Ultimately, his goal continues to be to elucidate how the world works from both a technical and a social perspective.

[This page intentionally left blank]

Table of Contents

List of Figures

List of Tables

[This page intentionally left blank]

List of Acronyms

ADCS	attitude determination and control subsystem
BSS	Boeing Satellite Systems
C&DH	command and data handling
CDC	Concurrent Design Center
CDF	Concurrent Design Facility
CE	concurrent engineering
CERI	Cognitive Engineering Research Institute
CIEL	Concurrent Integrated Engineering Lab
CPM	Critical Path Method
DoD	Department of Defense
DSM	Design Structure Matrix
DSMC	Defense Systems Management College
EPS	electrical power subsystem
ESA	European Space Agency
ESD	Engineering Systems Division
ESTEC	European Space Research and Technology Centre
EV	expected value
EXIX	EXcel Information eXchange
GSFC	Goddard Space Flight Center
HOQ	House of Quality
ICDF	Integrated Concept Development Facility
ICE	Integrated Concurrent Engineering
ICOM	input-control-output-mechanism
IDC	Integrated Design Center
IDL	Instrument Design Laboratory
IDSC	Interactive Design and Simulation Center
IMDC	Integrated Mission Design Center
INCOSE	International Council on Systems Engineering
ISAL	Instrument Synthesis and Analysis Laboratory

ISIS	IMDC System for Information Sharing
JPL	Jet Propulsion Laboratory
LaRC	Langley Research Center
LEO	low Earth orbit
MiSL	Mission Simulation Lab
MDL	Mission Design Laboratory
MTRL	Mission Technology Readiness Level
NASA	National Aeronautics and Space Administration
NDM	naturalistic decision making
PDC	Product Design Center
PERT	Program Evaluation and Review Technique
PRIME	Process Reasoning and Information Management Environment
PSE	power system electronics
QAP	quadratic assignment procedure
QFD	Quality Function Deployment
RPD	recognition-primed decision
SADT	Structured Analysis and Design Technique
SDSM	Sensitivity Design Structure Matrix
SE	systems engineering
SEM	systems engineering management
SMM	shared mental model
SESAC	Systems Engineering Services and Advanced Concepts
SSME	Space Shuttle Main Engine
STC	socio-technical congruence
TRL	Technology Readiness Level
TT&C	telemetry, tracking, and command
UAV	uninhabited air vehicle
UCAV	uninhabited combat aerial vehicle
WTRL	Weighted average Technology Readiness Level

Nomenclature

Q = modularity for a given community structure of a network

n = number of nodes in a network or rows/columns in a Design Structure Matrix

m = number of edges in a network or dependencies in a Design Structure Matrix

l = average distance between nodes in a network

C = clustering coefficient of a network

$S_{x,y}$ = mental model sharedness between team members x and y

$D_{x,y}$ = number of design drivers selected in common by team members x and y

D_x = number of drivers selected by team member x

D_y = number of drivers selected by team member y

$SMM_{x,y}$ = shared mental model between team members x and y

C_{SMM} = stability of shared knowledge

ΔS = change in shared knowledge

S_{pre} = pre-session average mental model sharedness in the team

S_{post} = post-session average mental model sharedness in the team

$I_{P,j}$ = perceived importance of design driver j

M = total number of team members selecting major design drivers

$C_{S\text{-}T}$ = socio-technical congruence

$N_{\#}$ = number of expected interactions reported by the team

N_b = number of unexpected interactions not reported by the team

N = total number of possible team interactions

F_{tot} = number of marks above the diagonal in a Design Structure Matrix

n_c = number of communities in a network

m_{zz} = number of edges entirely within the z^{th} community of a network

m_z = number of edges to or from any element in the z^{th} community of a network

[This page intentionally left blank]

Chapter 1

Introduction

This thesis is based on the simple premise that engineering is done by people – individuals each with their own knowledge, thoughts, and views about the system being designed. From this perspective, the research seeks to approach systems engineering in a new way. For the present purposes, systems engineering is not taken to be merely the technical integration of components and subsystems to optimize cost, performance, and schedule, nor is it even a multidisciplinary role that integrates the perspectives of various parties and seeks to deliver value to the stakeholders. While both of these functions are critical dimensions of what a systems engineer does, the goal of this research is to expand the definition of systems engineering as a holistic systems-level outlook that requires breadth and depth of knowledge about the system, the customer, and the varied thoughts and perspectives of the designers and developers of the system.

To accomplish this goal, systems engineering cannot be studied from a technical standpoint alone. Although the research presented in this thesis is necessarily grounded in a formal and rigorous analysis of the engineering design process, it also incorporates the findings of social and organizational psychology into the dynamics of the design environment. Based on this interdisciplinary perspective, the thesis culminates with a model of the engineering design process that takes into account both technical aspects of the design and the thoughts and interactions of the members of the design team.

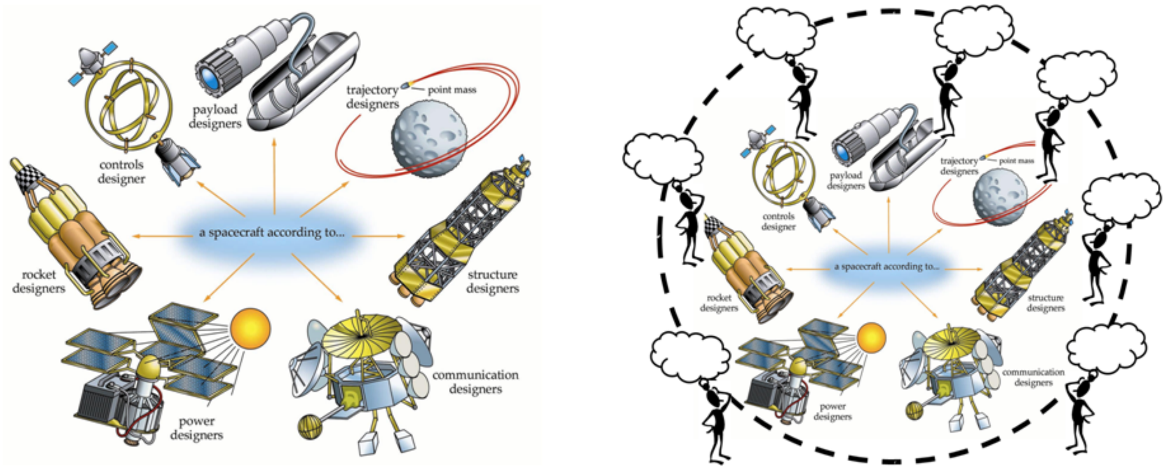
The purpose of this chapter is to introduce the motivation for and structure of the research. Section 1.1 explains the motivation for the research – to integrate the thoughts and opinions of individuals into the design process. Section 1.2 describes the three-part research approach, which consists of separate analyses of the technical and the social aspects of the design process and an interdisciplinary means of integrating them. Then, section 1.3 defines the problem to be addressed in terms of three research questions – a technical question, a social question, and an interdisciplinary question. Finally, section 1.4 describes the structure of the thesis and provides an overview of the contents of each chapter.

1.1. Research Motivation

The motivation for this research comes from the growing need to incorporate a large number of diverse professionals into the design, development, and deployment of complex engineered systems. Each of these people has a distinct and equally valid set of viewpoints and priorities that needs to be considered in the design process. These differing perspectives can be partially attributed to differing education and training, but often the differences are internal and individual. It is not enough to simply describe each engineer's perspective in terms of his or her function or discipline alone. To truly understand how each person involved in the process views the important issues in the design, it is necessary to directly incorporate the particular knowledge and thoughts of each individual as a person and not merely as a representative of a particular feature or subsystem.

The research presented in this thesis is intended to be broadly applicable across the design of all complex engineered systems, but the particular setting for which the model is constructed is the conceptual design phase of scientific spacecraft. Furthermore, the model is based on the design process in an Integrated Concurrent Engineering (ICE) environment. This rapid, collaborative design setting provides a valuable opportunity to study the space systems design process in an accelerated format so that several designs can be examined in a relatively short time. The ICE environment maps relatively well to a full development program. The difference, of course, is that each subsystem or discipline in ICE is represented by just one person. Still, as a laboratory for this research, the ICE design setting facilitates both the development of a systems-level model of the design process and an analysis of the dynamics in a team of engineers from a variety of backgrounds and disciplines.

Figure 1-1 describes the motivation for the research. The figure is based on a common perception that each person involved in the design process sees the system through the lens of his or her own function. Figure 1-1(a) depicts the complementary yet frequently conflicting priorities of the various subsystem and discipline designers involved in the process. For example, the communication designers see the spacecraft primarily as a collection of antennas, and the trajectory designers see it as a point mass in orbit.



(a)

(b)

Figure 1-1. Shared Knowledge among the Designers of a Complex System. (a) Perspectives on Spacecraft Design. From Robinson (2008). **(b) Perspectives on Spacecraft Design: The Role of Shared Knowledge.** Adapted from Robinson (2008).

The figure provides an insightful look at the tensions that exist between the priorities that need to be met in designing the various components of the system, but this type of depiction is complete only if it includes the human element. Figure 1-1(b) includes three modifications of the original representation that account for the role that people play in the design process. First, an image of a person is placed next to each discipline to indicate the simple fact that discipline engineers are people and not merely instruments of spacecraft design. Second, each person has a thought bubble to represent the individual views of each engineer that extend beyond the disciplinary perspective. Third and perhaps most importantly, a dotted line connecting all of the thought bubbles indicates that some overlap exists among the viewpoints of the engineers. Presumably, this overlap in viewpoints is necessary for the engineers to integrate their perspectives, resolve critical design trades, and produce a design that satisfies customer requirements and delivers value to all stakeholders.

The central theme of this thesis is this overlap among the thoughts and perspectives of the engineers and how it relates to the design process. In the literature, the common thoughts and viewpoints that exist among the members of a team have been analyzed from a variety of perspectives, including aggregation of pair-wise shared knowledge, quantitative inferences from team interactions, and qualitative analysis based on observation. Although the role of shared knowledge in the *operation* of complex systems has been discussed in the literature for several years (e.g., Rouse et al. 1992), little formal research has been done on the role of shared knowledge and cognition in the *design* of those systems. Still, the thought process of the engineers is undoubtedly an important aspect of the design process that warrants attention. The extent of knowledge sharing in some way must affect the outcome of the design, and the design process can, in turn, affect the shaping of shared knowledge. This is perhaps a truism about the nature of a creative endeavor like engineering design, but the form and nature of this socio-cognitive aspect of the design process remains a mystery. The goal of this research is not to resolve this broad and complex issue but rather to open a discussion on the role of shared knowledge and cognition in the engineering design process. The resulting model extends the existing literature on both product development process (e.g., Ulrich and Eppinger 2004) and shared knowledge in teams (e.g., Cannon-Bowers et al. 1993, Klimoski and Mohammed 1994), but it represents only a starting point in developing tools and methodologies for analyzing the design of complex engineered systems from a socio-cognitive perspective.

1.2. Research Approach

The research presented in this thesis is divided into three parts: a technical analysis, a social analysis, and an interconnection between them. The first part of the research is based on a model of information flow using parameter dependencies among all of the subsystems and disciplines involved in the space systems design process. This systems-level model of the technical design process is based on a matrix representation of inputs and outputs among all parameters that need to be computed over the course of the work. The analysis is done using the Design Structure Matrix (DSM), a matrix-based tool used to analyze complex product development processes (e.g., Eppinger 1991). This portion of the research leads to a set of insights about the design process life cycle and the way that the team can be organized to improve design outcomes.

The information flow model takes into account the variety of disciplinary perspectives involved, but it does not include the human element. To capture that aspect of the design, the second part of the research presents a network-based model of shared cognition in the team. Starting from each engineer's own perceptions of the important issues in the design process, a metric for measuring the overlap in two engineers' viewpoints is proposed, and a methodology for scaling the analysis to teams of any size is developed. Similar analysis has been done in other contexts (e.g., Lim and Klein 2006), but the model presented in this thesis also incorporates the time element in the measurement of shared cognition. The resulting metric of change in shared knowledge is related to various aspects of the system, including mission concept maturity, development time, launch mass, and system cost.

The purpose of the third part of the research is to integrate the first two parts to reveal interdisciplinary insights that cannot be gained from either analysis alone. This portion of the work examines the content of knowledge and cognition (i.e., the actual substance of the overlap among team members' thoughts and viewpoints), the specific role of each subsystem/discipline in the formation of that knowledge, and the connection between shared knowledge and the interactions among the engineers. Thus, this research addresses the previously untested but recently identified question in the literature on the role of shared knowledge in engineering design teams (see Badke-Schaub 2007).

The research presented in this thesis contributes to the growing body of knowledge in the emerging field of Engineering Systems. This field is based on the realization that engineered systems often are too complex to be understood in terms of purely technical analysis. Therefore, research in this field generally includes both technical and social dimensions that together facilitate a more complete understanding of complex engineered systems. This thesis is explicitly divided into a technical component (analyzing the design process) and a social component (modeling shared knowledge), and then these two parts are integrated into a third *socio-technical* component. The three research questions, discussed in the next section, follow directly from these three components of the work.

1.3. Problem Definition

The definition of the problem addressed in this thesis is structured around three research questions that map to the social, technical, and socio-technical components described in the previous section. The first question deals with the technical design process. Specifically, it addresses the particular challenges and opportunities that come with developing a systems-level representation of the early conceptual design phase in a fast-paced collaborative atmosphere. Thus, the first research question is

Q1: ***How can the Design Structure Matrix be used to analyze and improve the process in a rapid collaborative design environment?***

The second question focuses on the social aspects of the design process and on shared knowledge in engineering design teams specifically. As will be discussed later in the thesis, most prior work in this area has been concerned mostly with static measurements of shared knowledge in small teams taken in a controlled laboratory setting or anecdotal evidence of real-world teams. The goal of this portion of the thesis is to build on that prior work by offering a *quantitative* model of shared knowledge in *engineering design* that incorporates *time-dependence* and is *scalable* to teams of any size. Accordingly, the second research question is

Q2: ***How can a network-based approach reveal the dynamics of shared knowledge in engineering design teams?***

The scope of this thesis, however, reaches beyond independent analyses of the technical design process and the socio-cognitive aspects of team dynamics. The overall purpose of the research is to integrate these two analyses to understand the relationship between the technical and the social. As stated at the beginning of this chapter, the research aims to broaden the standard definition of systems engineering to include the people whose knowledge and effort make the design and development of complex engineered systems happen. This can be accomplished only by an integrative approach that directly incorporates the analysis of team dynamics into the modeling of the design process. The third and final research question, which captures this essential interdisciplinary component of the thesis, is

Q3: ***What is the relationship between the design process and shared knowledge in engineering systems design? ?***

In Chapter 8, the discussion will return to these three questions, and an answer will be provided for each based on the analysis presented in the intervening chapters.

1.4. Structure of the Thesis

Broadly, the thesis can be divided into three parts, summarized in Table 1-1. The first part, which includes Chapters 1-3, frames the research, reviews the relevant literature, and describes the organization in which the research was conducted. The second part, which is made up of Chapters 4-6, focuses on data analysis, model construction, and presentation of the results. Chapters 4, 5, and 6 correspond to the technical, the social, and the integrative socio-technical portions of the research, respectively. The final part, which is composed of Chapters 7 and 8, concludes the thesis, offers recommendations, and discusses the possibilities for future work. The contents of each of the eight chapters of the thesis are as follows.

Chapter 1 introduces the research motivation, the framework around which the research is based, and the definition of the problem in terms of three research questions. Chapters 2 and 3 describe the *domain* and *context* of the research, respectively. Chapter 2 provides an overview of the literature relevant to the research. Because of the interdisciplinary nature of the thesis, the literature review covers a broad array of fields, including systems engineering, product development, human factors engineering, and organizational and social psychology. Chapter 3 offers a primer on space systems design in general and in the Integrated Concurrent Engineering (ICE) design environment in particular. The chapter is divided into four sections that discuss space systems design, the ICE environment in general, the particular ICE design center in which the research was done, and the process of data collection in this design setting.

The next three chapters focus on data analysis, results, and model development. Chapter 4 makes up the technical component of the research and addresses the first research question, Q1. This chapter focuses on the specialized approach taken to construct a Design Structure Matrix representation of the design process in the ICE environment and the insights that can be gained from applying DSM-based analysis in such a context. The analysis reveals the phases of the design life cycle, the interdisciplinary design trades, clusters of interdependent disciplines based on those trades, and the starting assumptions that can be made to optimize the process.

Chapter 5 describes the social component of the research and addresses the second research question, Q2. In this chapter, a network-based model of shared knowledge in engineering design teams is developed. The model is tested by demonstrating a relationship between the dynamics

Table 1-1. Structure of the Thesis.

Part	*Purpose*	*Chapter*	*Chapter Content*
1	Framing the Research	1	Introduction
		2	Research Domain
		3	Research Context
2	Analysis and Modeling	4	Technical Analysis
		5	Social Analysis
		6	Socio-Technical Analysis
3	Discussion and Conclusions	7	Implications
		8	Contributions

of shared knowledge over time and technical attributes of the system being designed. The chapter includes a detailed sensitivity analysis to rigorously evaluate the usefulness of the model.

Chapter 6 integrates the findings presented in Chapters 4 and 5 and provides interdisciplinary insights about the design process. This chapter forms the integrative socio-technical portion of the research and provides answers to the third research question, Q3, along several dimensions. The analysis in this chapter begins with a discussion of the relationship between the dynamics of shared knowledge and the product of the design. Then, the discussion turns to the content of shared knowledge in the team and those disciplines whose design outcomes play a particularly important role in the process. Next, the change in shared knowledge over time is related to team coordination. The chapter closes with a discussion of the value of the shared knowledge construct for analyzing the work of engineering design teams.

Chapters 7 and 8 together conclude the thesis. Chapter 7 focuses on the implications of the results for existing and potential future ICE laboratories. The chapter offers a series of insights and recommendations framed around the four elements of the design center studied: people, process, tools, and facility. In that chapter, a comprehensive, data-driven standardized model of the design process is presented, and the applicability of the research to larger organizations and enterprises is discussed. Chapter 8 provides a concise summary of the results, explains the contributions to the literature and to the field of Engineering Systems, discusses the limitations of the research, and offers a set of suggestions for future work. Ultimately, this thesis is intended to begin the discussion on the relationship among shared knowledge, process, and product in engineering design. The results are exciting and promising, but they only scratch the surface of possible ground-breaking research in this new interdisciplinary area of study.

Chapter 2

Literature Review

The purpose of this chapter is to review the relevant literature on which the research presented in this thesis is based. The review draws on several distinct bodies of literature that together reflect the interdisciplinary nature of the thesis. Still, all of this diverse literature pertains directly to the problem of analyzing and improving the design of engineered systems by addressing both the technical and the social aspects of the process. Taken together, the literature reviewed in this chapter reveals a growing need and opportunity to incorporate the findings of social and organizational psychology into systems engineering.

The literature discussed in this chapter comprises the theoretical and practical *domain* of the research, leaving references most directly related to context and methodology to other parts of the thesis. The topics discussed in this chapter are drawn from the wide array of academic disciplines to which this research contributes: systems engineering, product development, human factors engineering, and organizational and social psychology. Chapter 3, on the other hand, includes references to works related to the *context* of the research, i.e., the specific type of engineering design environment in which data were collected. The references that were consulted specifically to support the development of the research *methodology* are included as needed throughout Chapters 4, 5, and 6.

The structure of this chapter is as follows. Section 2.1 discusses the ongoing debate regarding the definition of systems engineering and discusses the role that people play in systems engineering. Section 2.2 offers a description of certain relevant aspects of the product development organization: the design process and the structure of the organization. This section includes an overview of tools and techniques for design process analysis, briefly reviews the existing literature on the particular technique used in this research, and then discusses ongoing research on the relationship between product architecture and organizational structure. Next, section 2.3 reviews the literature on groups and teams with a particular emphasis on expertise and functional diversity. After that, section 2.4 provides an overview of the role of knowledge and cognition from several distinct perspectives. Finally, section 2.5 synthesizes the literature and explains the opportunity for an integrative analysis on the connection between engineering design process and shared knowledge in the design team.

2.1. What is Systems Engineering?

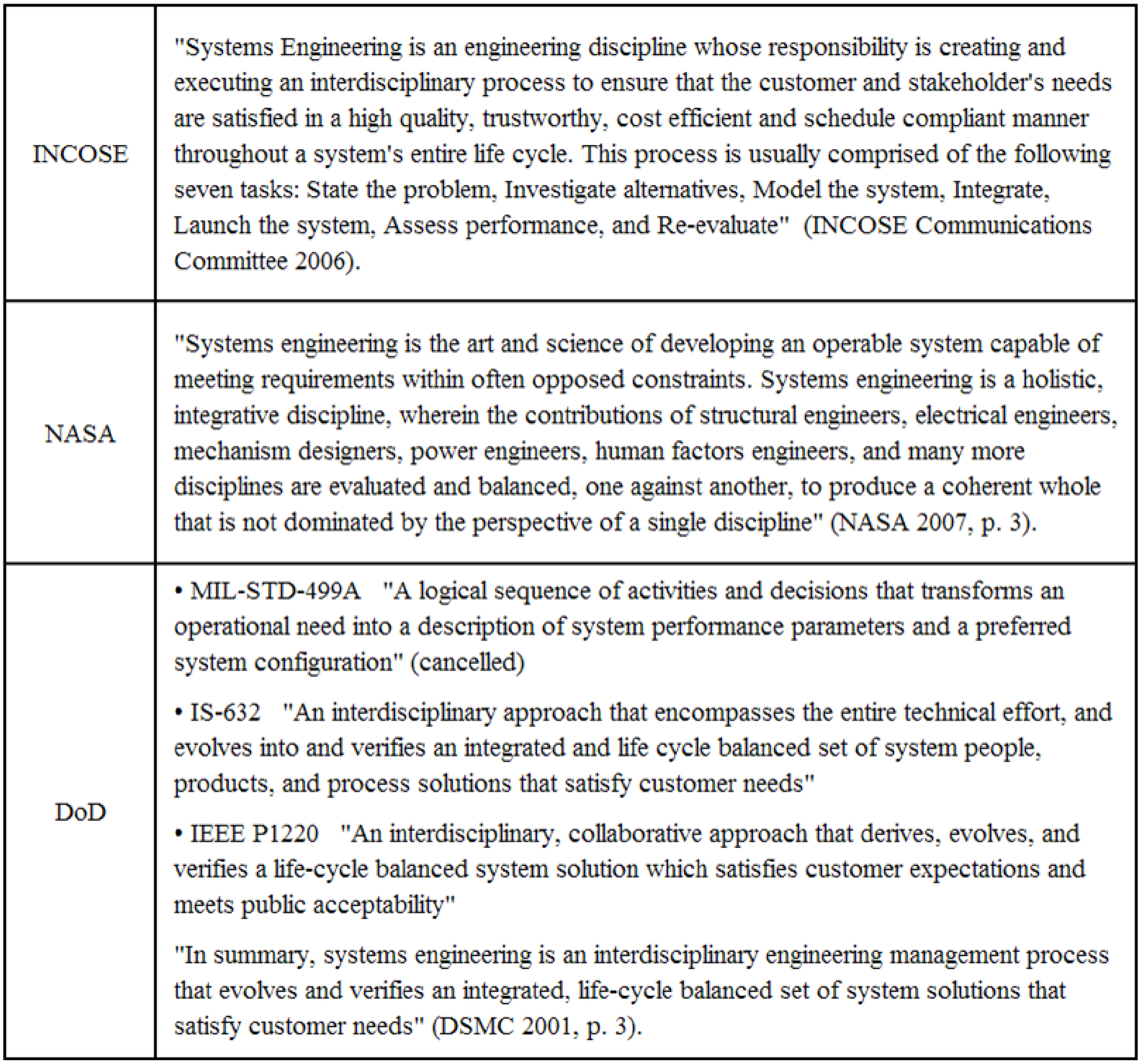
This section introduces the variety of perspectives on systems engineering in the literature. In the first subsection, the many existing definitions of systems engineering are reviewed. In the second subsection, the roles of the various people involved are described.

2.1.1. Definitions of Systems Engineering

Systems engineering (SE) is a somewhat elusive concept that is generally "fraught with controversy" (Martin 1997, p. 3). At times, it can seem as though there are as many definitions of systems engineering as there are people involved in the activity. Many books have been written describing how systems engineering is done. These books often begin with a list of definitions for the term as interpreted by a variety of organizations and authors (e.g., Buede 2000, Blanchard 2008). Sage (1992) opens his well-known treatment of the topic with three different definitions that depend on the perspective taken. According to the structural definition, SE is "management technology." The functional definition holds that SE is a "combination of theories and tools, carried out through use of a suitable methodology and set of systems management procedures." Lastly, the purposeful definition states that the role of SE is "information and knowledge organization" (p. 10). According to these definitions, one might view SE alternatively as a technology, a process, or a philosophy.

The National Aeronautics and Space Administration (NASA) and the Defense Systems Management College (DSMC) have produced entire documents outlining what SE is and how it should be done in NASA and the Department of Defense (DoD), respectively (NASA 2007, DSMC 2001). In an attempt to establish a baseline definition, the International Council on Systems Engineering (INCOSE) has reached a consensus that incorporates the perspectives of many senior systems engineers from a variety of organizations (INCOSE Communications Committee 2006). The full definitions used by NASA, the DoD, and INCOSE are provided in Table 2-1.

Table 2-1. Definitions of Systems Engineering in Practice.



INCOSE	"Systems Engineering is an engineering discipline whose responsibility is creating and executing an interdisciplinary process to ensure that the customer and stakeholder's needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system's entire life cycle. This process is usually comprised of the following seven tasks: State the problem, Investigate alternatives, Model the system, Integrate, Launch the system, Assess performance, and Re-evaluate" (INCOSE Communications Committee 2006).
NASA	"Systems engineering is the art and science of developing an operable system capable of meeting requirements within often opposed constraints. Systems engineering is a holistic, integrative discipline, wherein the contributions of structural engineers, electrical engineers, mechanism designers, power engineers, human factors engineers, and many more disciplines are evaluated and balanced, one against another, to produce a coherent whole that is not dominated by the perspective of a single discipline" (NASA 2007, p. 3).
DoD	• MIL-STD-499A "A logical sequence of activities and decisions that transforms an operational need into a description of system performance parameters and a preferred system configuration" (cancelled) • IS-632 "An interdisciplinary approach that encompasses the entire technical effort, and evolves into and verifies an integrated and life cycle balanced set of system people, products, and process solutions that satisfy customer needs" • IEEE P1220 "An interdisciplinary, collaborative approach that derives, evolves, and verifies a life-cycle balanced system solution which satisfies customer expectations and meets public acceptability" "In summary, systems engineering is an interdisciplinary engineering management process that evolves and verifies an integrated, life-cycle balanced set of system solutions that satisfy customer needs" (DSMC 2001, p. 3).

According to the INCOSE definition, systems engineering is an engineering discipline. NASA, on the other hand, presents a broader view, referring to SE as an art and a science whose purpose is to integrate a variety of perspectives and priorities. The DoD definition, which is a combination of three prior standards of what SE is, explicitly highlights the organizational aspects of systems engineering and thus frames it as an *engineering management process*. From the DoD perspective, systems engineering management (SEM) includes not only the process but also the broader outlook that includes integrating these perspectives and planning for the entire life cycle.

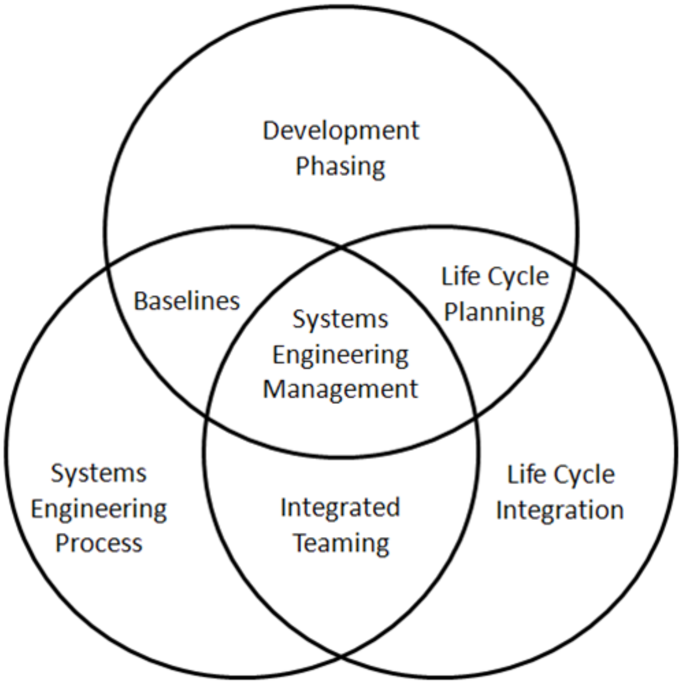


Figure 2-1. Three Activities of Systems Engineering Management. Adapted from Defense Systems Management College (2001, p. 4).

Figure 2-1 depicts the DoD view of systems engineering management (DSMC 2001). The principles shown map to various aspects of this thesis. The three large circles representing *systems engineering process*, *life cycle integration*, and *development phasing* are essentially the core components of Integrated Concurrent Engineering (ICE), the design environment on which the present research is based. The intersection between each pair of circles highlights a key element of design that is at the core of this research. In the ICE environment, the goal is to rapidly produce *baselines* that achieve mission objectives. This is accomplished through integrated *life cycle planning*, which involves all relevant disciplines so that every aspect of the entire mission life cycle from conception to disposal can be considered. The most important feature of the ICE environment is the real-time give-and-take among all discipline engineers by *integrated teaming*. One of the basic goals of this thesis is to examine the role of the people in such integrated teams. This examination of the people and the process thus leads to new insights about *systems engineering management*.

2.1.2. People in Systems Engineering

Many of the definitions of systems engineering cited in the previous subsection refer to the customer as an essential part of what systems engineering is. Generally, the primary goal of systems engineering is to satisfy customer needs. Ulrich and Eppinger (2004) provide a five-step procedure to systematically determine the customer's needs, while Blanchard and Fabrycky (2006) highlight the customer's responsibility to specify high-level requirements. Ultimately, the customer is the primary driver of the system's design. The customer is not necessarily always right, but he or she is paying for the system and therefore dictates what it ought to do (Eisner 2008). Thus, the fundamental objective of the design process is to deliver value to the customer. This is more important than simply decreasing cost or improving performance along some metric. Delivering value to the customer requires an explicit recognition of the fundamental tension between facts and perceptions. In any system design, the facts are not nearly as important as the customer's perceptions of those facts (Maier and Rechtin 2002). In general, effective design requires open communication between the design team and the customer (Buede 2000).[1]

Maier and Rechtin (2002) offer a systematic approach to identifying the critical stakeholders. Borrowing from socioeconomic research, they present a framework called "the four whos" (p. 79). As the name implies, the key stakeholders can be identified by answering four questions: "who benefits? who pays? who provides? and, as appropriate, who loses?" (p. 80). A notable aspect of this framework is the explicit separation of the customer or client (who pays) from the user (who benefits). Therefore, using the more general term "stakeholder" captures the complex array of people interested in the system's development at a variety of levels (Buede 2000). For the spacecraft studied in this research, the federal government (and thus the taxpayer) often pays for the design, development, and operation of the system, but the user is normally a team of scientists that may or may not be publicly funded. In most cases, however, the data returned from the missions are made available for the use of the entire scientific community.

---

[1] The type of system most immediately relevant to this discussion is large-scale and generally has a specific user and/or customer with an interest in certain specific design decisions. This is appropriate in this thesis since the focus is on complex engineered systems. Of course, for consumer goods, the process occurs somewhat differently. Customers and users are important stakeholders in those cases, but the means of meeting their needs is less direct and explicit.

Cameron et al. (2008) developed a technique for systematically mapping interactions among these various stakeholders. The result is a block-flow representation in which blocks and flows represent stakeholders and the movement of resources among them. In the space systems example to which this framework is applied, eight types of stakeholders are identified: science, security, international partners, economic actors, Executive and Congress, the general public, educators, and the media. In addition, the model includes six types of flows: policy, money, workforce, technology, knowledge, and goods and services. The core of the analysis involves tracking the loops in the network. These loops usually consist of many types of flows, indicating that resources are converted to different forms in the process of delivering value to each of the various stakeholders.

Ultimately, though, one set of stakeholders remains unaddressed in all of these works. The engineers responsible for the design and development of the system seem to be taken as exogenous to the stakeholder framework. While it might be argued that this group fits into the economic stakeholder group, this categorization would only account broadly for the existence of jobs in the aerospace sector. It does not, however, incorporate the diverse perspectives of the members of the design team. Given the importance of teams in systems engineering as highlighted in the previous subsection, it would appear that the knowledge, views, and thoughts of the individuals involved in the design ought to be integrated into the definition of systems engineering. Yet none of the definitions include cognition of the design team as part of what SE is or what SE does. This thesis attempts to demonstrate the need to include this vital component into the common parlance of systems engineering. Following an interdisciplinary review of the relevant literature, the synthesis at the end of the chapter demonstrates the need for research focused on this issue.

2.2. Product Development: Design Process and Organizational Structure

Product design and development is a growing field that deals with all aspects of creating a product and bringing it to market. Although this broad field includes a variety of topics, the focus of the present discussion is on those particular aspects that are relevant to this thesis. Specifically, these topics are design process analysis and organizational structure. In the first subsection, a variety of traditional tools and techniques for process analysis are reviewed. In the second subsection, the particular tool chosen for process analysis in this thesis, the Design Structure Matrix (DSM), is introduced. Finally, in the third subsection, several studies on the relationship between product architecture and organizational structure are discussed.

2.2.1. Traditional Tools and Techniques for Design Process Analysis

In the past few decades, several types of global representations of the design and development of complex engineered systems have been devised. The purpose of this subsection is to review four established tools and techniques often used to manage complex product development processes. The representations discussed are the Gantt chart, the PERT chart, the Structured Analysis and Design Technique (SADT), and Quality Function Deployment (QFD). Examples of these four tools are shown in Figure 2-2.

The Gantt chart and the PERT (Program Evaluation and Review Technique) chart are relatively simple representations of the timeline for a project. The Gantt chart provides a straightforward representation of the timing of all tasks that must be completed. It does not, however, provide any information about the dependencies among tasks (Ulrich and Eppinger 2004). The PERT chart resolves this drawback of the Gantt chart by explicitly showing the flow of work that leads from one task to the next. Each task is represented by a box that includes information such as task start date and expected completion time. A series of arrows connecting the boxes depict the effects of each task on subsequent tasks. The structure of the PERT chart facilitates a relatively simple calculation of the critical path, the flow of tasks through the chart that takes the longest to complete. The use of the PERT chart for this purpose is often called the Critical Path Method (CPM). This tool is useful for planning the execution of an established activity, but it assumes that backflows of information (feedback) and repetition of work (rework) do not occur (Eppinger et al. 1992). For this reason, the PERT chart is best used for managing well-defined development and integration activities but has limited use at the highly iterative conceptual design stage.

The Structured Analysis and Design Technique (SADT) is similar to PERT but expands on it in two important ways. First, it considers more than one type of dependency among tasks. The boxes in SADT documents are called ICOMs, which stands for input-control-output-mechanisms. Using this convention, various conditions and rules (controls) and required resources (mechanisms) are included along with inputs and outputs (Santarek and Buseif 1998).

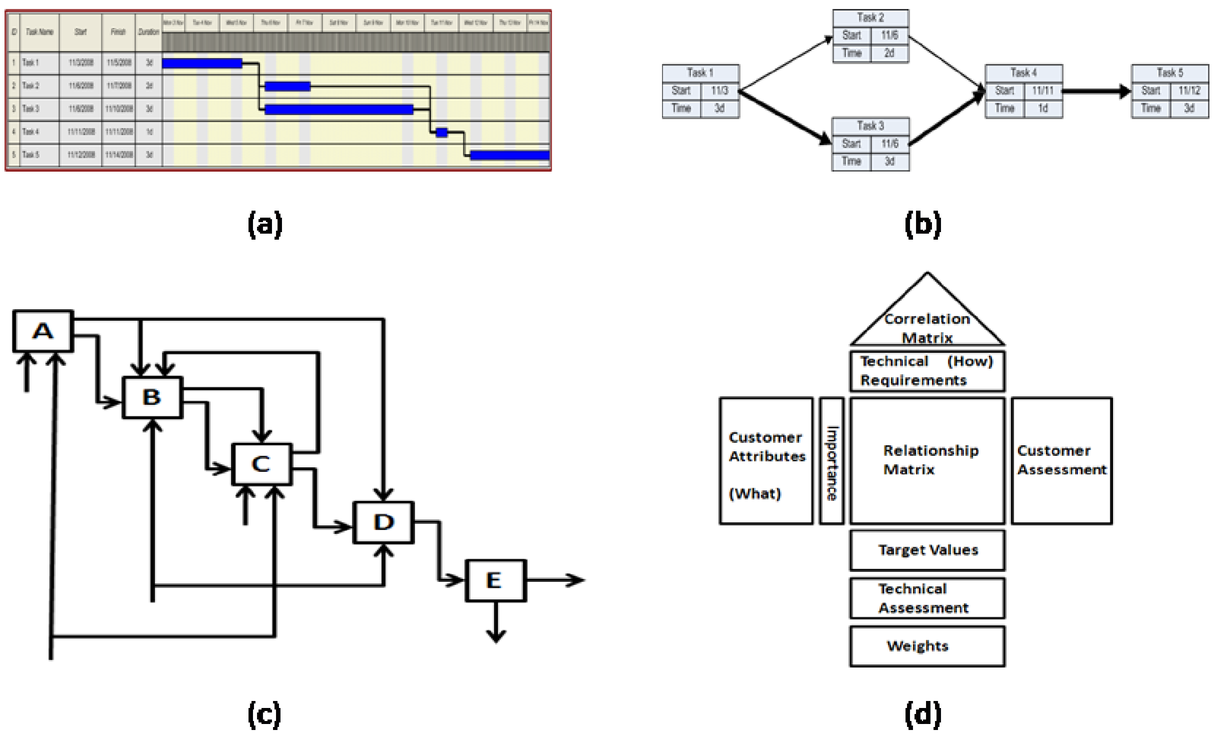


Figure 2-2. Simple Examples of Four Design Process Analysis Tools. (a) Gantt chart. (b) PERT chart. Arrows representing work flow along the critical path are shown in bold. **(c) SADT document.** Adapted from Santarek and Buseif (1998) **(d) House of Quality.** Adapted from Temponi et al. (1999).

Second, unlike PERT charts, SADT documents account for the complexities of feedback and rework by allowing flows to return to previous tasks. Thus, SADT is able to capture the richness and detail of the design process in a way that most other tools cannot. Nevertheless, the depiction of every task in a complex project in terms of boxes and arrows quickly becomes far too cumbersome for systems-level analysis. As a result, SADT documents often are nearly as complex as the processes that they model, and the method provides little more than descriptive capacity with limited potential for process improvement (Eppinger et al. 1992).

Quality Function Deployment is a procedure for mapping customer requirements to product characteristics through the use of a tool called the House of Quality (HOQ). In this technique, a series of HOQs successively maps several levels of detail (e.g., technical requirements and component characteristics) to the next until the path from customer needs to operational steps can be identified (Temponi et al. 1999). Although this main function of QFD distinguishes it from the other three techniques just discussed, it also has a secondary function of depicting dependencies among individual tasks in the “roof” of the HOQ. Because QFD uses a matrix-based format, its overall size and complexity scales slowly with the complexity of the project as compared to an equivalent network representation. For this reason, it is more accessible and provides a clearer picture of the overall design than does SADT. Its ability to represent information flow, however, is limited because the dependencies are catalogued as correlations in a triangular half-matrix that does not provide any indication of the direction of dependence among tasks. For this reason, QFD cannot be used to analyze the global effects of interdependencies among individual tasks. The Design Structure Matrix, on the other hand, is intended for that purpose and thus is the method chosen for design process analysis in this thesis. The next subsection briefly describes the DSM and reviews some of the important literature on its development and past uses. A full discussion of the DSM and its advantages is provided in Chapter 4.

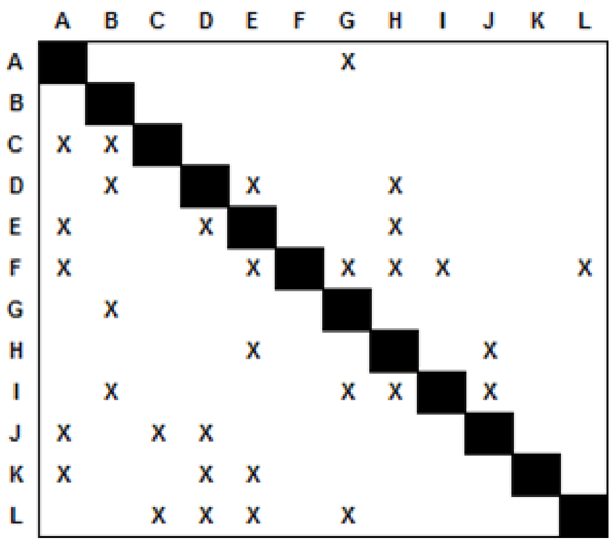
2.2.2. The Design Structure Matrix

The Design Structure Matrix is a matrix-based representation of interdependencies in a system. According to this modeling technique, a mark in a cell of the matrix means that the item in the row requires information from the item in the column as an input. This can be seen in the example DSM shown in Figure 2-3. In general, a DSM can be built as one of four types: component-based, team-based, activity-based, and parameter-based. Browning (2001) offers a full review of the DSM literature, including a detailed description of the usage of each type. He draws on a comprehensive collection of prior research to provide an analysis of the relationships among the DSM types and the barriers to their use in the real world. Essentially, the basic difference among them is the nature of the dependencies, which has important implications for the analysis procedures used and for the insights that can be gained about the systems that they represent.

The DSM methodology was first proposed by Steward (1981a, 1981b), but the literature on the concept remained sparse for the following decade. One possible reason for this is that the computational resources needed to conduct DSM-based analysis on any but small idealized systems simply were not available to individual researchers. In the early 1990s, however, Eppinger and several colleagues published a series of works reintroducing the methodology and proposing some computational algorithms to implement DSM-based analysis (Eppinger et al.

1990, Eppinger 1991, Gebala and Eppinger 1991). Further studies then offered applications to real-world systems such as automobile parts and semiconductors.

In one such study, Eppinger et al. (1994) built and analyzed a DSM for a component of an automobile's powertrain. Based on the analysis, they identified four coupled groups of tasks that mapped to the design teams within the program studied. In a related study, the same authors also conducted a DSM-based analysis of an automotive brake system (Eppinger et al. 1992, Eppinger et al. 1994). This work revealed a design process characterized by three distinct blocks of work: (1) a series of upfront tasks, such as obtaining customer requirements, that could be completed sequentially; (2) a set of tightly interdependent tasks making up the most difficult portion of the work; and (3) another set of mostly sequential tasks focused on settling the final details of the design. In another study on the design of the camshaft, the authors found that the most iterative portion of the work was the design phase (Eppinger et al. 1992).



	A	B	C	D	E	F	G	H	I	J	K	L
A	■						X					
B		■										
C	X	X	■									
D		X		■	X			X				
E	X			X	■			X				
F	X				X	■	X	X	X			X
G		X					■					
H					X			■		X		
I		X					X	X	■	X		
J	X		X	X						■		
K	X			X	X						■	
L			X	X	X		X					■

Figure 2-3. Example of a Design Structure Matrix.

In addition to the automotive studies just described, Eppinger et al. (1994) also built and analyzed a DSM for the design and development of a semiconductor. This study identified several groups of tightly coupled tasks, and many of the groups overlapped with each other, indicating that tasks across groups required close coordination. In addition, the analysis showed that certain other sets of tasks could be completed simultaneously because they were not dependent on each other at all. Finally, the analysis also indicated the existence of feedback from the end of the development cycle back to the beginning. The authors termed this phenomenon "generational learning feedback" (p. 10) because the applicable lessons were not learned in time to improve the current product but could be applied to the next generation of the product line (Eppinger et al. 1994). These and other related early works paved the way for researchers from several communities to conduct studies that used the DSM to reach actionable insights about a variety of complex systems.

James Rogers of NASA Langley Research Center (LaRC) applied the DSM formalism to conceptual aircraft design (Rogers 1999). Using an optimization algorithm capable of "minimizing the time, cost, feedback couplings, and crossovers of each iterative subcycle" (p. 272), this work showed that reordering the tasks in the aircraft design process can lead to improvements of more than 80% in both design cycle time and cost. At the same time, however, the analysis also revealed that optimization reduced the number of opportunities to carry out separate processing tasks concurrently. This tradeoff between optimization and parallel processing necessarily must be addressed when determining task order in a process characterized by large subcycles (Rogers 1999). As will be discussed later in the thesis, the DSM constructed for the present research contains such a large subcycle. Instead of determining task order by trading among various attributes, though, the approach taken here is to determine the starting assumptions that can be made to enter the loop in the first iteration.

Browning and Eppinger (2002) extended the DSM methodology by incorporating uncertainty into the framework. In this work, they constructed two DSMs to represent the design process for an uninhabited combat aerial vehicle (UCAV). In the first DSM, each cell contains a probability that rework will be necessary. In the second DSM, each cell contains a measure of the impact of rework. This DSM-based representation is then incorporated into a broader model that uses discrete event simulation to study the effect of process architecture (the layout of the design process) on cost and schedule risk. The output of the model, which is a cost and duration probability distribution for sets of inputs to the process, facilitates a comparison among different ways of structuring the work.

Kalligeros et al. (2006) applied a similar variant called the Sensitivity DSM (SDSM) to oil production facilities. In the SDSM, the cells of the matrix do not indicate just the existence of a dependency but also the extent of that dependency. Thus, a particular cell shows how much one variable changes as a result of a change in another. For this reason, the SDSM is necessarily less populated than a traditional DSM because some pairs of variables may have some interdependence but low sensitivity. Using this methodology, Kalligeros et al. used the SDSM to identify product platforms, i.e., a common baseline system architecture on which variations can be constructed.

One of the basic goals of this thesis is to construct and analyze a DSM for the full space mission design process. Previous work in this area exists but is somewhat limited. Before the

DSM came into wide use, Padula et al. (1989) had constructed a matrix-based representation of the notional design of an experimental space system. The analysis of this matrix revealed groups of interdependent tasks that could be used to organize the design. More recently, Ahmadi et al. (2001) used the DSM framework to optimize the activities in the conceptual design of turbopumps for the Space Shuttle Main Engine (SSME). The latter study was based on a detailed analysis but focused on just a small part of the overall system design. The former work, on the other hand, was done using a DSM-like representation of the full system under consideration. Still, that system did not include certain elements of a full space mission, such as operational and ground elements, and the analysis was conducted when many software tools for DSM analysis were not yet available. Thus, this thesis expands on those prior studies by conducting a full DSM-based analysis of an entire space mission design. Furthermore, the thesis takes this analysis a step further by incorporating the interactions of the design team. To address this aspect of the work, the next section reviews the literature exploring the connection between organizational structure and product architecture.

2.2.3. Organizational Structure and Product Architecture

Ulrich and Eppinger (2004) define product architecture as “the scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact” (p. 165). The architecture of a given product can be conceptualized in terms of its modularity. In a completely modular product, all chunks are entirely self-contained and have simple, well-defined interfaces among them. On the opposite end of the spectrum, a completely integral architecture is one in which the chunks are completely connected to each other and distinctions among them are virtually undetectable. In reality, of course, most products are not strictly modular or integral but rather are characterized by a certain level of modularity. This, of course, implies that modularity/integrality can be quantified based on certain system properties. To do this, Hölttä-Otto and de Weck (2007) proposed a DSM-based metric of modularity. They tested both their metric and another one previously proposed by Guo and Gershenson (2003) on two pairs of products: (1) landline and cellular telephones and (2) desktop and laptop computers. In addition, they applied their metric to several previously built DSMs for others products. Based on this work, they showed that products whose designs are driven primarily by technical performance requirements tend to take on more integral architectures whereas those driven by business requirements tend to be more modular. For a review and evaluation of several measures of modularity that have been proposed in the literature, see Van Eikema Hommes (2008).

Product architecture has some important implications for various attributes of the product and the associated process, including ease of product change, product variety, component standardization, product performance, manufacturability, and product development management (Ulrich and Eppinger 2004). Because the focus of this thesis is on the dynamics of the design team, the most important of these implications for the present purposes is the last one – product development management.

Taking a broad view on product architecture, Langlois (2002) used the concept of modularity to frame the reasons for the existence of firms in the economy. Essentially, his argument rests on the notion that the "nonmodularity" (i.e., integrality) within a firm enables individuals to leverage their collective knowledge to improve the firm's products. Baldwin and Clark applied this idea to a specific case study, the history and architecture of IBM's System/360 (1997, 2000). They found that IBM was able to create a superior product through modular architecture but that, in so doing, it "also weakened substantially the forces that previously had kept whole computer systems within the boundaries of one firm" (2000, p. 212). This facilitated the emergence of many firms that specialized in modular components that could be plugged into System/360 without any direct IBM involvement. As a result, "the industry began to change in structure as well: it began to evolve into its present form — a highly dispersed *modular cluster* of firms" (2000, p. 213). Following the dot com crash, however, the authors found that they needed to revisit their theory. The resulting advice to managers was to retain a strong understanding of the precise nature of their product's modularity and to understand the blurred lines between process and product. They also noted that modular design facilitates frequent change and that this effect is most pronounced in the case of small hidden modules with high technical potential (Baldwin and Clark 2001).

Gulati and Eppinger (1996) combined an extensive review of the literature with their own accompanying field study of audio system development at a large automobile manufacturer. This effort resulted in a series of general insights about the connection between product architecture and organizational design. Their findings include the following:

- "Decomposition determines team assignments" (p. 12).
- "Incidental interactions catalyze the formation of problem solving teams" (p. 13).

- “Architecture determines communication patterns” (p. 14).
- “Architecture determines the feasibility of co-location” (p. 15).
- “Static organizations give rise to static architectures” (p. 17).
- “Organizational skills and capabilities affect architecture” (p. 18)
- “Supplier relationships can affect architecture” (p. 18)
- “Organizational design of a globally distributed team affects architecture” (p. 20).

Based on these outcomes, the authors propose that the mechanism enabling the connection of architecture to organization is a two-part process involving problem decomposition and subsequent system integration.

McCord and Eppinger (1993) discussed ways of resolving this issue, which they termed the “integration problem.” Specifically, they used the term to refer to the need for many separate teams focusing on different aspects of the design of a complex system to integrate those pieces into a final product. In their application to the development of an automobile engine, the authors found that delineations between existing product teams could be modified to improve integration efforts. The results of this study indicated that the broader system teams (made up of several product teams) could be reorganized in a way that would allow certain overlaps among them to facilitate integration. In addition, the study concluded that an integration team consisting of the most integrative product teams could be created to focus exclusively on issues affecting the global product architecture.

In addition to the above studies on broad organizational issues, many researchers have done work examining the relationship between product architecture and technical communication in a product development organization. Allen (1985), for example, has shown that the probability of regular technical communication between two team members depends on the physical distance between them. Specifically, the probability of communication increases as separation distance decreases, and the probability reaches an asymptotic low point at relatively short distances (about 30 meters). Moreover, Allen and Henn (2007) found that this relationship is true not only for face-to-face communication but also for telecommunication media such as the telephone. Morelli et al. (1995) examined communication in organizations in terms of expected versus “actual” (as reported in surveys) interactions. Studying a team developing electrical connectors, the authors constructed two matrices of communication in the team. First, they built a DSM to describe expected information flow. Then, they used weekly surveys to construct a network of reported communication in the team. Their results indicated that frequent interactions were more predictable than infrequent ones and that two-way communication was predicted more accurately than one-way communication.

Sosa et al. (2003) used the DSM to identify modular and integral subsystems of a commercial aircraft engine. Along with the DSM, they constructed a team interaction matrix using survey data on frequency and importance of communication between design teams in the organization. Overlaying these matrices provided a systems-level view of the predicted and reported interactions in the organization. Based on a statistical analysis of this mapping, the authors determined some important differences between the communication patterns of modular and integrative subsystems. Specifically, they found that more unpredicted interactions occurred during the design of modular systems than in the design of integrative systems, that integrative teams are "more effective at overcoming the barriers imposed by organizational boundaries" (p. 250), and that different types of interfaces tend to be handled in the design of modular and integrative systems.

Cataldo et al. (2008) constructed a similar study in a distributed software development project. As in the two previous studies discussed, the authors constructed two types of matrices, which they called a Coordination Requirements matrix and an Actual Coordination matrix. Coordination Requirements were derived from a combination of engineers' task assignments and syntactic dependencies in the source code. Actual Coordination was based on four different types of interactions or sources of interaction in the organization: structural, geographical, modification requests, and Internet Relay Chat. They then compared required to actual communication using a metric called socio-technical congruence (STC), which they defined simply as "the proportion of coordination activities that actually occurred … relative to the total number … that should have taken place" (Cataldo et al. 2008, p. 5). Based on an overall analysis using all four types of coordination, they concluded that a high level of STC resulted in a 32% reduction in the time taken to resolve modification requests.

In this thesis, the concept of socio-technical congruence is used to operationalize the relationship between product architecture and organizational structure. Because the present research was not conducted in a distributed setting but rather with a collocated and tightly integrated team, the proposed metric is somewhat different from the one used by Cataldo et al. (2008). Furthermore, because conceptual spacecraft design does not result in a final developed product with measurable performance, STC is instead compared to various technical attributes of the system being designed. This analysis also involves an aspect of team coordination and organizational structure that has not yet been considered in this chapter – the role of knowledge and cognition in the design team. To address this critical albeit intangible aspect of the design, the next two sections focus on prior research on teams. Section 2.3 discusses existing research on expertise and functional roles in teams, and section 2.4 provides an overview of research on shared knowledge and cognition done from both a social science and an engineering perspective.

2.3. Expertise and Functional Diversity in Teams

This thesis is closely related to the extensive literature on organizations, their form and structure, and the interactions among the people within them (see, for example, Allen 1985, Galbraith 1994, Krackhardt and Hanson 1993, March and Simon 1993, Schein 1996). The present research contributes to that broad body of literature but focuses on one subset of organizations: the role and dynamics of teams. A team can be defined as any group of individuals that share a common identity and work together toward a common goal. The defining feature that separates a team from any group of people is its high degree of task interdependence (Salas et al. 2004).

The interdependent nature of teams is a central aspect of the team-based design setting studied in this thesis. During each design session, the members of the team work together in a single facility for a well-defined, intensive period to produce a conceptual design of a scientific spacecraft. The customer for each design session invests significant resources to achieve the best possible design that meets the stated objectives. For this reason, it is essential that the design team consist of some of the most highly trained and experienced engineers available, i.e., *experts* in their disciplines. In addition, the purpose of the design center is to produce a full design that accounts for all parts of the system and all phases of the mission life cycle. Thus, each engineer represents a separate function and brings a distinct set of skills to the team. In other words, the team is characterized by a high degree of *functional diversity*. The many forms of diversity (e.g., race, ethnicity, gender, age, level of education, etc.) comprise an important part of the literature on teams (e.g., Jackson et al. 1991, Jehn et al. 1999, Phillps and Loyd 2006). For the purposes of this thesis, however, the focus is specifically on expertise and functional diversity in teams.

Salas et al. (2006) argue that an "expert team" is more than just a team of experts. To capture this, they define an expert team as

> *a set of interdependent team members, each of whom possesses unique and expert-level knowledge, skills, and experience related to task performance, and who adapt, coordinate, and cooperate as a team, thereby producing sustainable and repeatable team functioning at superior or at least near-optimal levels of performance (Salas et al. 2006, p. 440).*

Based on a review of the literature on the topic, the authors compiled a list of the primary characteristics of expert teams. According to their synthesis, the members of expert teams (1) share mental models (a concept that will be discussed in the next section of this chapter), (2) learn and adapt, (3) self-correct, (4) have well-defined but not rigid roles, (5) share a common purpose, (6) have skilled leadership, (7) have a high degree of trust and confidence in the team, (8) achieve the best possible performance with minimal errors, and (9) effectively coordinate and communicate (Salas et al. 2006).

If an expert team is not simply a team of experts, then a more intricate framework of high-performing teams is needed to identify and create expert teams. Based on a wide array of prior work, Ancona et al. (2002) developed a theory of what they call X-teams. These teams are characterized by five distinguishing features: external activity, extensive ties, expandable tiers, flexible membership, and mechanisms for execution. The authors concluded that X-teams tend to perform better than more static and inwardly-focused (i.e., "traditional") teams. This, however, does not imply that X-teams constitute the answer to increased performance under all circumstances. The authors suggest that a team should follow the X-team model if it operates within a flat organizational structure, it works with complex and dynamic information, and/or its work is interdependent with activities occurring outside of the team.

Gruenfeld et al. (2000) studied the effect of "worldliness" on team performance. This concept includes aspects of three of the key characteristics of X-teams: external activity, extensive ties, and flexible membership. The authors found that the ideas offered by "itinerant" members (those who switched teams temporarily and later returned to their original teams) were less likely to be used directly by the team following their return. On the other hand, they also found that the presence of itinerant members tended to influence the indigenous members to generate ideas. Although the direct effect of worldliness on the team was less than expected, it had a significant indirect effect through improved performance of the other members. This result is consistent with the finding of Carroll et al. (2006) that expertise diversity based on prior departmental affiliations led to greater depth and creativity.

Ancona and Caldwell (1992) examined functional diversity, a central aspect of many expert teams (including the ones studied in this thesis). In their study, the authors considered two types of function-related diversity: tenure diversity (the extent to which members joined the team at different times) and professional diversity (the mix of education and experience among members). They found that while tenure diversity is associated with performance, professional diversity can have a mix of positive and negative effects. Based on these results, they speculate that, for certain teams, professional diversity might bring "more creativity to problem solving and product development, but it impedes implementation because there is less capability for teamwork than there is for homogeneous teams" (Ancona and Caldwell 1992, p. 321). Thus, this work suggests that professional diversity is beneficial if and only if the team establishes mechanisms to leverage the benefits while mitigating the drawbacks.

Bonner et al. (2002) showed that small teams (of three members) rely on the expertise of the highest performing team member. This study, however, dealt with only one potential area of expertise in the team. The authors point out that their ranking of individual performance, which was based on previous tasks completed separately by each team member, would not be as applicable in a setting where multiple domains of expertise are relevant. In addition, the subjects in this study were told who the top performing member of the team was rather than being left to determine that during the task. Littlepage and Mueller (1997) created a scenario in which this type of information was not available to the team a priori. Their results indicate that expert knowledge is more likely to be utilized by the team if the expert is extraverted and talkative and if he or she demonstrates expertise through reason and logical argument.

Similarly, Jackson (1996) found through a survey of the teams literature that judgments made by only one person tend to be ignored either because the person does not have the confidence to express a differing opinion or because the team lacks confidence in the uncorroborated view. As a result, Jackson suggests that decision-making and problem-solving in teams could be improved by ensuring that there is some intersection among the members' areas of expertise. According to the results of a study by Stasser et al. (1995), however, this overlap might not be necessary in all cases. They determined that explicit identification of an individual as having an expert role in a given area can significantly increase the likelihood that the person's view is vocalized and incorporated into team discussion.

In an effort to understand the role of overlapping expertise, Marks et al. (2002) studied the effect of cross-training on team performance. They defined three levels of cross-training: positional clarification (having information about other team members' roles), positional modeling (having both information and the opportunity to observe other team members), and positional rotation (actively participating in other team members' roles). Their model indicates that cross-training leads to shared knowledge, which in turn results in increased coordination (timing of interdependent steps) and backup (assisting other team members). Coordination and backup processes then result in increased team performance. Still, this chain of relationships was not perfect because some uncertainty arose in the degree of influence that cross-training has on the formation of shared knowledge. Since the present thesis is largely based on the role of shared knowledge in a team of experts working in an engineering design context, the next section explores the literature on shared knowledge and cognition both in the types of settings discussed above and in engineering design.

2.4. Knowledge and Cognition in Engineering and Organizations

It is perhaps a truism that the members of a team must share certain knowledge about their work. At the same time, however, most people undoubtedly have heard of groupthink, the phenomenon that arises when the desire for group preservation outweighs the need for effective decision-making. As Cannon-Bowers et al. (1993) note, shared knowledge can become a liability if it creates such a high level of cohesiveness that "the desire for unanimity overrides realistic appraisal and consideration of possible courses action" (p. 236). Nevertheless, groupthink is merely the possible down side of a broader phenomenon in teams. In reality, the precise cost-benefit tradeoff associated with common viewpoints among team members is still an open question in the literature. One of the fundamental goals of this thesis is to contribute to that debate by demonstrating the existence of a relationship between shared knowledge and the engineering design process.

The purpose of this section is to explore the broad and diverse literature on shared knowledge and cognition from several distinct perspectives. First, the conventional literature on shared mental models among small teams working mostly in controlled laboratory settings is reviewed. Then, some of the key principles in the field of real-world, or naturalistic, decision-making are discussed. After that, a brief overview of the user-centric engineering discipline of human factors is provided, and the cognitive aspects of this work are highlighted. Finally, the section closes with an argument for the relevance of shared knowledge and cognition to engineering design teams.

2.4.1. Shared Mental Models

The concept of a mental model is used to describe the way in which an individual perceives his or her environment. Klimoski and Mohammed (1994) note that some authors refer to a mental model as a catch-all for any knowledge about a given environment while others use the term only to describe organized knowledge that helps one to "understand phenomena, make inferences, and experience events by proxy" (p. 405). Rouse and Morris (1986) define mental models more specifically as "mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions of future system states" (p. 351). This view of mental models as "mechanisms" leads to a definition of "sharedness" of mental models. Two people can be said to hold a shared mental model (SMM) if they utilize mechanisms that lead to similar descriptions, explanations, and predictions of the system.

Still, the debate over how to operationalize shared mental models is ongoing. Klimoski and Mohammed (1994) argue that the shared mental model concept is more appropriately viewed as a valuable and meaningful construct than a simple metaphor. In other words, they hold that it is a measurable variable and not merely an abstract notion. In the literature, the extent of similarity between mental models is often quantified on some numerical scale. Rouse et al. (1992), however, take a pragmatic approach to the use of mental models, stating that the construct should add value and not simply be a way of labeling knowledge. According to their view, "measurement of mental models is a process of identifying an intervening construct that may not be unique but does provide a consistent and useful explanation of the data of interest" (Rouse et al. 1992, p. 1304).

To help ensure the utility of shared mental models, a distinction is often drawn among different types of mental models, normally based on their underlying content. Cannon-Bowers et al. (1993) classify mental models into two categories: task mental models (those that facilitate accomplishing a task) and team mental models (those that allow each individual to work effectively as a member of the team). In two related studies, Mathieu et al. (2000) and Mathieu et al. (2005) drew an empirical distinction between the notions of team and task mental models in two-person teams, or dyads. To operationalize SMMs, they computed a score based on participants' common perceptions of the relationships among several task and team attributes. In Mathieu et al. (2000), the authors found that sharedness of team mental models was positively related to team performance and that the relationship was fully mediated by team processes, though this result was not reproduced by Mathieu et al. (2005). Also in Mathieu et al. (2000), the authors found that sharedness of task mental models was not strongly correlated with performance, but such a relationship was found in Mathieu et al. (2005). The authors attribute at least some of the differences to a larger sample size used in the latter study (Mathieu et al. 2005). In both studies, however, the authors found that task mental model sharedness was strongly correlated with team processes (Mathieu et al. 2000, Mathieu et al. 2005).

Kameda et al. (1997) conducted a study to measure the effect of "cognitive centrality" in teams of three, or triads. To do this, they constructed what they call a sociocognitive network and proposed a measure of cognitive centrality in the network. Their results indicated that cognitively central members have greater influence in the team than do cognitively peripheral members. In addition, they found that unshared knowledge held by just one team member has a minor effect on group decisions. Lim and Klein (2006) took this idea of examining cognition in larger teams a step further. Based on a field study of 71 seven- to eight-person air combat teams, they devised a means of measuring shared knowledge in the entire team. To measure pair-wise shared mental models, they took an approach similar to that of Mathieu et al. (2000) and Mathieu et al. (2005). Instead of stopping at dyads, however, they computed the average level of sharedness among all possible pairs of team members. In addition, they computed the same metric for each team member against experts' responses on the task as a measure of mental model accuracy. They found statistically significant correlations among all five measures examined: taskwork mental model similarity, teamwork mental model similarity, taskwork mental model accuracy, teamwork mental model accuracy, and team performance. They did not, however, find any evidence of interaction between team mental model similarity and accuracy. Contrary to expectations, the effect on performance of sharing a mental model does not clearly change as a result of the mental model's accuracy.

The above study made an important contribution by extending the discussion on shared mental models to larger teams, but the approach taken was still a simple mean-based aggregation of the individual pair-wise shared mental models in the team. It did not provide a means of analyzing the broader effects throughout the team. Langan-Fox et al. (2004) highlight this difficulty in extending the notion of a shared mental model between two people to the larger construct of a *team mental model*, which they refer to explicitly as "a synergistic functional aggregation of the [team's] mental functioning representing similarity, overlap, and complementarity" (p. 335). Klimoski and Mohammed (1994) take a similar view, referring to a team mental model as "an emergent characteristic of the group, which is more than just the sum of individual models" (p. 426).

The next subsection briefly discusses a body of literature that directly addresses the need for a new view of team mental models based on the broader dynamics of the entire team. Much like the work of Lim and Klein (2006), this research involves large teams and is done in field settings. Rather than constructing a metric of sharedness, however, the researchers base their work on the notion that the thinking of the team as a whole is best documented through qualitative and anecdotal methods.

2.4.2. Naturalistic Decision Making

Gary Klein and colleagues have made valuable contributions to the literature on team cognition, specifically within a field called naturalistic decision making (NDM). This work is relevant to the current thesis because it deals with experienced professionals making decisions subject to ill-defined goals, missing or ambiguous information, and high time pressure (Klein 1998). Based on qualitative observations of firefighters and other teams meeting those criteria, Klein developed the recognition-primed decision (RPD) model. A key part of this model is the notion of mental simulation. A mental simulation is essentially a type of a mental model in which "a decision maker cognitively constructs a model and sets it in motion to see what happens" (Klein and Crandall 1995, pp. 333-334). This allows the decision maker to visualize possible decisions and to see them through before actually making a decision.

Extending the research in naturalistic settings to teams, Klein (1998) has developed the notion of the "team mind" as an analogy for the way that a team grows and acts. Based on anecdotal evidence of actual working teams, he identified the features indicating that a team operates as though it has a mind. Like an individual mind, a team mind has some basic functions: working memory, long-term memory, limited attention, perceptual filters, and learning. Similarly, teams develop certain capabilities much like children. These capabilities include competencies, identity, cognition, and metacognition (Klein 1998).

In addition, Klein et al. (2003) consider the role of "macrocognition," the high-level mental processing that occurs during complex real-world activities. In contract to microcognitive functions like solving a puzzle, macrocognitive processes take place in naturalistic environments and include skills like planning and dealing with uncertainty. The authors note that this type of thinking generally takes place in collaborative environments. In addition, they argue that macrocognition should play an important part in research on cognitive systems engineering, a human-centered approach to the design of engineered systems. The next section reviews some of the literature in this area of study.

2.4.3. Psychological Aspects of Human Factors Engineering

Classical human-machine systems engineering, or more simply *human factors engineering*, is the branch of engineering that deals with issues of humans-in-the-loop in the design of complex engineered systems. Traditionally, this field has focused on physiological concerns and skill-related human behaviors. Over the past couple of decades, however, the role of the psychological aspects of human-machine interactions has received increased attention. This sub-discipline of human factors engineering that directly incorporates cognition into the design is called *cognitive ergonomics* (Sage 1992) or *cognitive systems engineering* (Woods and Hollnagel 2006). Hollnagel (2003) has compiled a collection of studies on theories, methods, and cases in *cognitive task design*.

Nancy Cooke of Arizona State University and the Cognitive Engineering Research Institute (CERI) and several colleagues have done pioneering work in *cognitive engineering*, a discipline focused on the design of systems for human use.[2] Kiekel and Cooke (2004) explain the

---

[2] http://www.cerici.org/

advantages of conducting human factors research through the lens of team cognition. They argue that studying the team as a thinking entity can improve design, intervention capabilities, and training. Cooke and Gorman (2006) discuss the different perspectives that can be taken to the measurement of team cognition. They refer to measurement techniques involving an average of pair-wise shared mental models, such as the method used by Lim and Klein (2006), as the *collective* approach to team cognition. In contrast, they take a *holistic* (or ecological) perspective based on the notion that team cognition is best operationalized by directly measuring team interactions. Using this approach, Gorman et al. (2004) observed 11 three-person teams that each engaged in seven uninhabited air vehicle (UAV) missions. They found in the first three missions that team communication became progressively more concise on average and that the team used a wide range of words in their interactions. Beginning with the fourth mission, however, the researchers noticed a point at which the results shifted discontinuously. At that point, the language used by the team became significantly more concise and less variable. This shift corresponded to a point at which the rapid rate of increase in average performance slowed. At that point, average performance began to approach an asymptotic limit.

Despite the orientation of cognitive engineering toward incorporating user cognition into design, the work in that area does not address the role of cognition in the design process itself. This thesis, on the other hand, is directly concerned with shared knowledge in engineering design teams. To address this topic, the next subsection discusses the applicability of shared knowledge to the engineering design process.

2.4.4. The Applicability of Shared Knowledge to Engineering Design

As this section has shown so far, a great deal of research has been done on the problem of shared knowledge and cognition in teams. Although some of this work is strongly grounded in engineering design, its focus is generally on how design is to be done when taking the cognitive dimensions of the *user* as a technical requirement. The role of cognition and shared knowledge among the engineers that actually design the systems, however, has been largely unaddressed in the literature. Still, collaboration and information exchange are essential aspects of the design process. For this reason, one of the main goals of this thesis is to develop a methodology for analyzing the role of shared knowledge and cognition among the designers of complex engineered systems.

Yang and Ji (2007) have begun work in an area of research based on the thoughts of engineers working on complex design projects. Using text-based analysis of conversation transcripts, they have developed a means for determining the probability that the team will choose each option among a group of design alternatives. Based on data from the same design setting, Ji et al. (2007) developed a model that uses speech patterns to identify the design team's overall preferences among alternatives. This promising research is akin to the holistic approach to shared knowledge (Cooke and Gorman 2006) that extrapolates team cognition from behavior. While this work essentially applies the holistic approach to small teams of engineers, an opportunity still exists to also measure shared mental models, the basic unit of analysis in the collective approach, among the members of teams working in a design context.

Badke-Schaub et al. (2007) have examined the applicability of shared mental models to the engineering design context. They argue that the concept of shared mental models can be valuable for studying design teams, but they note some important differences from the teams that have been studied in prior research. Whereas most research on shared mental models has focused on tactical teams with clearly defined objectives, engineering design involves creative teams with a high degree of autonomy. They point out that a design team's task is subject to a much higher level of uncertainty than are the tasks of an operational team. For design teams, it is often the case that

> *there is no definitive formulation of the problem and there is not [a single] best solution to the problem. The consequence is that team members have to develop a common model in order to use existing knowledge and to guide new information rather than following regular operations like the standard operating procedures in flight control (Badke-Schaub et al. 2007, p. 17).*

This statement has two important implications for the use of the shared mental model construct in studying design teams. First, it shows that the metrics for shared mental models must be viewed somewhat differently for design teams. Second, it points out that "team members have to *develop* a common model" (emphasis added). In other words, the time element is an important aspect of mental models in design. The specific adjustments to the mental model construct proposed for engineering design will be discussed in detail in Chapter 5 of the thesis. For now, the next section offers a synthesis of the literature and highlights those works that form the foundation for the present research.

2.5. Synthesis of the Literature

The goal of this section is to evaluate, analyze, and synthesize the literature reviewed in the preceding sections. Because the thesis consists of three parts, the synthesis of the literature follows the same structure. The first subsection discusses the work in product development and design process analysis on which the technical analysis in this thesis builds. The second subsection synthesizes the existing literature on shared knowledge and introduces a new method for quantitatively analyzing shared mental models in teams of any size. Finally, the third subsection discusses the opportunity addressed in this thesis to integrate the bodies of literature on design process and shared knowledge and thus provides the overall motivation for the research.

2.5.1. Design Process Analysis

In recent years, the specialized approach of the Integrated Concurrent Engineering design environment has been used not only as a means of producing full conceptual designs in a short time but also as a laboratory for analyzing the space mission design process. For example, Olson et al. (2009) developed a multiagent simulation model to analyze the design process of Team X at the Jet Propulsion Laboratory (JPL). At one level, this thesis extends the work in that area by developing an alternative methodology for ICE design process analysis using the Design Structure Matrix. As the first DSM representation of the ICE design process, the model presented in this thesis provides an important practical and theoretical contribution by integrating these two distinct approaches to complexity management in engineering design. McCord and Eppinger (1993) demonstrate that the DSM can be used to identify concurrency in the design process but note that this reveals the complementary problem of determining ways to integrate the outcomes of the parallel work. The ICE environment, on the other hand, provides a means of explicitly enabling this integration at the conceptual design level (Karpati et al. 2003, Sercel et al. 1998). The present research uses the DSM to identify concurrency in that integrated process and, in so doing, presents an analysis that combines concurrency with integration.

More broadly, the process analysis model constructed in the ICE environment is also the first DSM representation of the full space mission design process that includes all phases of the mission life cycle and utilizes all of the basic procedures for DSM analysis. Padula et al. (1989) built an early high-level DSM-like representation of a particular type of space system, and Ahmadi et al. (2001) used the DSM to conduct a more detailed analysis of one component of another type of space system. This research extends these works by providing a DSM representation that captures the design process for a full space mission, including aspects like mission operations, reliability, integration, and costing. Because the ICE environment includes all of the relevant disciplines and thus is representative of the full space mission design process, it offers a useful laboratory for constructing a baseline DSM for that type of design, which would be prohibitively complex to do from scratch. Although the ICE-based DSM would require certain modifications on a case-by-case basis, it provides a template systems-level representation that can be applied to the design and development of space systems more generally.

2.5.2. Shared Knowledge and Cognition

One of the primary goals of this thesis is to develop a model of shared knowledge that is applicable to the real-world engineering design context. In this chapter, three perspectives on shared knowledge and cognition have been addressed: *naturalistic*, *collective*, and *holistic*. A brief summary of these three approaches is provided in Table 2-2. The approach to shared knowledge developed in this thesis is intended to combine the advantages of the other three.

The naturalistic approach offers three distinct advantages: real-world teams can be analyzed in detail, the entire team can be viewed as a thinking entity with a "team mind" (Klein 1998), and the time element can be incorporated via mental simulation (Klein and Crandall 1995). In the collective approach, on the other hand, the analysis is quantitative, and the focus is on small teams of two to three members working on a well-defined task in a controlled laboratory environment (e.g., Mathieu et al. 2000). Lim and Klein (2006) aggregated shared mental models in a larger real-world team by taking the average among all dyads. Whether analyzed at the dyad level or in aggregate, these works usually frame shared mental models as a static property of the team that has some effect on process and performance. The advantages of the collective approach are that the analysis is based on actual cognition of the team members and that it can be scaled to larger teams.

Table 2-2. Approaches to Shared Knowledge in Teams.

Approach Type	*Research Setting*	*Data Type*	*Analysis Method*	*Unit of Analysis*	*Team Size*	*Example from the Literature*
Naturalistic	Field	Qualitative	Anecdotal Accounts	Team	Any	Klein (1998)
Collective	Usually Lab But Can Be Field	Quantitative	Common Views (Aggregated)	Dyad	Any, Aggregated From 2 Members	Lim and Klein (2006)
Holistic	Lab	Quantitative	Counts of Spoken Words	Team	3 Members	Cooke et al (2004)

The holistic approach is based on the notion that "cognition" of a team as a whole is manifested in the behavior of the entire team. Thus, like the naturalistic approach, it views the team as a thinking entity. It is generally applied to three-person teams conducting well-defined operational tasks, and it offers a quantitative means of analyzing team cognition (e.g., Gorman et al. 2004). The primary advantage of this approach is that it provides a global view of cognition in the team using quantitative analysis. It also has some capability to include time-dependence through a comparison of multiple runs over time, though it does not incorporate how knowledge develops within the timeframe of a single project.

This thesis addresses an opportunity to combine the three approaches discussed above by building and analyzing a network-based representation of shared mental models in teams. The resulting model of shared knowledge is quantitative, dynamic, and scalable. Like the collective approach, the model constructed in this research includes a direct measure of pair-wise shared mental models. Instead of simply aggregating these results, however, shared knowledge is treated as an emergent property of the entire team, which follows from the basic of philosophy of the holistic approach. Team cognition is represented here not by computing a simple average of shared mental models but rather by using social network analysis (Wasserman and Faust 1999, Newman 2003) to measure team-wide shared knowledge from the structure of the relationships. Like the naturalistic approach, this model can be used to analyze shared knowledge in large real-world teams because social network analysis is intended to represent large groups of entities.

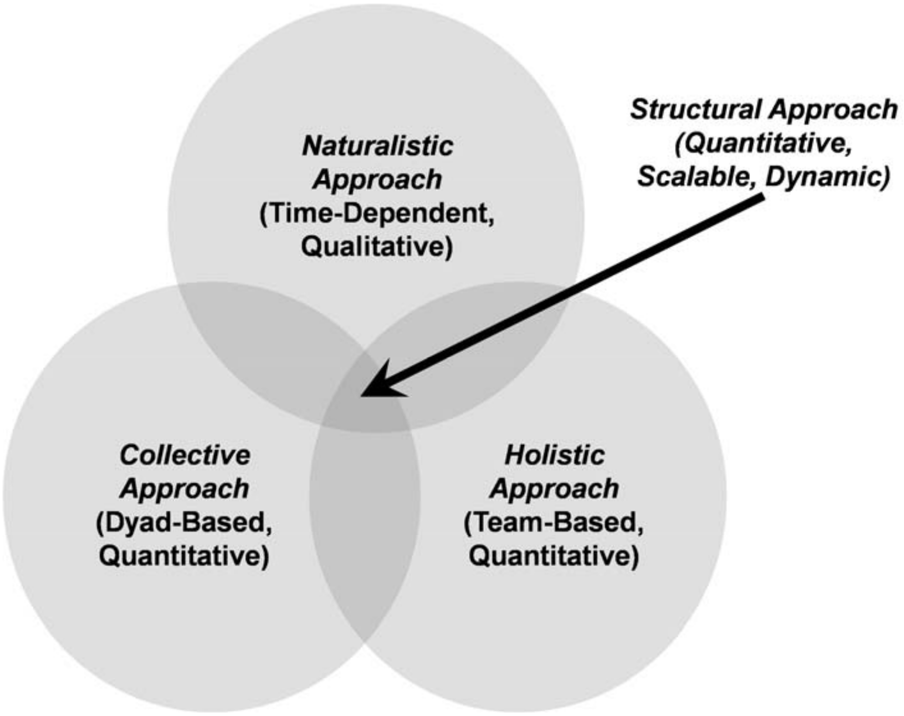


Figure 2-4. The Structural Approach to Shared Knowledge. The structural approach is quantitative, can be scaled to teams of any size, and incorporates time-dependence in the model.

Also following on the naturalistic approach, the proposed model incorporates the time element into the analysis of shared knowledge. Specifically, it includes a means of measuring the change in shared knowledge over time. This is done by calculating the structural similarity of pre-work and post-work team mental model networks. Because this integrative network-based model represents a new perspective on shared knowledge based on the structure of relationships among team members' shared mental models, it is termed the *structural* approach to shared knowledge. This proposed approach and its relationship to the three existing approaches are depicted in Figure 2-4.

2.5.3. Socio-Cognitive Analysis of Engineering Systems Design

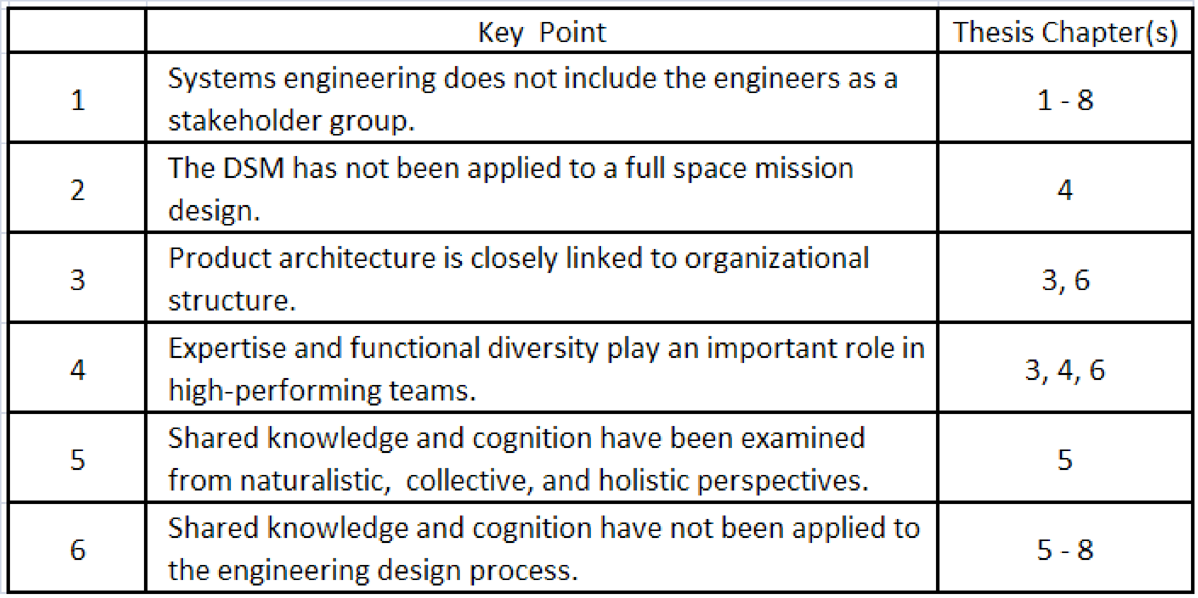
The overall goal of this thesis is to understand the relationship between the engineering design process and shared knowledge in the design team. Because the integrated approach used has been developed for this research, the literature on which it is based is essentially the intersection of the works discussed in the previous two subsections. Those works and the discussion throughout this chapter have highlighted several key points that come from each of the academic disciplines addressed by the research. These points are as follows:

1. Systems engineering addresses the role of stakeholders in the process but does not count the engineers that design the system among those stakeholders.
2. The DSM offers valuable design process analysis capabilities but has not been applied to a full space mission design.
3. Product architecture is closely linked to organizational structure.
4. Expertise and functional diversity play an important role in high-performing teams.
5. Shared knowledge and cognition have been examined from several perspectives, including naturalistic observation, quantitative collective aggregation of pair-wise shared mental models, and measurement of team behavior from a holistic standpoint.
6. Shared knowledge and cognition have *not* been applied to the engineering design process.

This research is intended to explicitly address and integrate these six points. Point 1 acknowledges an issue that exists in systems engineering, while point 6 highlights an opportunity to contribute to the literature that directly addresses that issue. Points 3 and 4 provide the reasons that the issue matters from a product and a process standpoint, respectively. Finally, points 2 and 5 together offer an analytical means by which the issue can be addressed. These points and the thesis chapters in which they are most directly addressed are listed in Table 2-3.

This chapter has reviewed and synthesized the literature that forms the basis of the research presented in this thesis. Chapters 4, 5, and 6 describe the methodology and analysis used to address the research problem. Before that work is presented, though, the next chapter first offers an overview of the context of the research, i.e., the particular setting in which the data were collected and to which the results can be most directly and immediately applied.

Table 2-3. Key Points from the Literature and Associated Thesis Chapters.



	Key Point	Thesis Chapter(s)
1	Systems engineering does not include the engineers as a stakeholder group.	1 - 8
2	The DSM has not been applied to a full space mission design.	4
3	Product architecture is closely linked to organizational structure.	3, 6
4	Expertise and functional diversity play an important role in high-performing teams.	3, 4, 6
5	Shared knowledge and cognition have been examined from naturalistic, collective, and holistic perspectives.	5
6	Shared knowledge and cognition have not been applied to the engineering design process.	5 - 8

[This page intentionally left blank]

Chapter 3

Research Setting: The Integrated Concurrent Engineering Design Environment

In the previous chapter, the background literature that forms the basis for this thesis was presented. This literature exists at the intersection of several fields but is all directly related to the fundamental problem of modeling the role of shared knowledge in engineering design. The literature reviewed in that chapter represents the *domain* of the research. This chapter serves the complementary role of introducing and describing the setting, or *context*, in which the research is conducted and the literature associated with it. The context for this research can be understood at three levels. Broadly, the research focuses on the conceptual design of scientific spacecraft. More specifically, the subject of the analysis is the Integrated Concurrent Engineering (ICE) design environment, and the particular ICE center in which the data were collected is the Mission Design Laboratory (MDL) at NASA Goddard Space Flight Center (GSFC).

This chapter is divided into four sections presented in increasing degree of specificity. First, the necessarily integral architecture of space systems design in general is explained. Then, the structure of the ICE environment and the history of its development are presented. After that, the Mission Design Laboratory, including its organizational context and its structure, is described. Finally, the process of data collection in the MDL is explained. This discussion includes the design sessions observed and the method of data collection used. The chapter then concludes with a brief introduction to the data analysis that will be discussed in the following three chapters.

3.1. The Integral Architecture of Space Systems

Since the dawn of the Space Age in the 1950s, the size, complexity, and importance of space systems have continued to grow. Although space-based technology began as a Cold War effort to demonstrate military strength, space systems have become critical not only for defense but also for telecommunications, navigation, science, and exploration. Despite the diverse purposes of these systems, they share certain important features in common. They all are complex and highly integral yet also consist of a set of certain well-defined subsystems and other disciplines that must be considered together in the design process. Furthermore, as discussed in Chapter 1, the design of each of these subsystems and disciplines comes with its own priorities that both complement and conflict with the others. Since this thesis is concerned with understanding the interactions among these functions, a general overview of space systems design in general is an important part of describing the context in which the research takes place.

Although the precise naming and characterization varies by mission type and organization, space systems generally have several components in common. According to Wertz and Larson (1999), these fundamentals of space missions are orbit and constellation; command, control, and communications architecture; mission operations; a space element, consisting of the payload and the spacecraft bus; a ground element; a launch element; and the subject of interest (e.g., a target of scientific investigation). In addition, the spacecraft bus consists of several subsystems, including the attitude determination and control subsystem (ADCS); telemetry, tracking, and command (TT&C), also called communications; command and data handling (C&DH); the electrical power subsystem (EPS); the thermal subsystem; the structures and mechanisms subsystem; guidance and navigation; propulsion; and computer systems.

These well-defined distinctions among the different subsystems and disciplines, however, belie the highly integral nature of the process and the product. In the design of any space system, mass is always at a premium because of the expense and difficulty associated with overcoming Earth’s gravity. As a result, the various subsystem engineers continuously make design trades to minimize the overall cost and mass of the system. For example, the work of ADCS can be accomplished in one of several ways, such as gravity gradient stabilization, i.e., taking advantage of differences in the force of gravity at different points on the spacecraft; spin stabilization; or three-axis stabilization using torquers, reaction wheels, thrusters, or other methods (Eterno 1999). Gravity gradient and spin stabilization have direct implications for the structure of the spacecraft, whereas three-axis stabilization affects the need for propellant tanks and thrusters in the propulsion subsystem.

Similarly, the spacecraft orbit is closely related to TT&C, EPS, and thermal. If the spacecraft spends a significant amount of time out of direct sunlight (or, for planetary spacecraft, on a trajectory toward the outer planets), the need for energy storage in the form of batteries or for power sources other than solar (e.g., nuclear) increases. At the same time, solar flux affects spacecraft and payload temperature (McMordie and Panetti 1999), so the thermal design is also dependent on the orbit. In addition, the spacecraft orbit affects opportunities for contact with ground stations by TT&C. Furthermore, TT&C makes trades between antenna aperture and transmitter power (Kirkpatrick 1999), which affects the requirements on EPS.

These are just a few of the interconnections that exist among the various subsystems and disciplines involved in the space systems design process. Not only do these technical components of the system interact closely and frequently, but the needs of the users and operators of the final system also need to be considered throughout the entire design life cycle. As Wertz and Larson (1999) note,

> *To explore a concept successfully, we must remove the walls between the sponsor, space operators, users or customers, and developers and become a team (p. 10).*

According to this statement, the developers (and designers) are not merely agents whose purpose is to deliver value to the customer. While that is certainly their ultimate objective, the engineers have their own thoughts and view about the design as well. Thus, this statement, which refers to the engineers and the customers as part of a team, captures one of the important goals of this thesis – to redefine systems engineering to include the perspectives of the designers and developers in the process.

The remainder of this chapter focuses on a design environment in which such an integrated developer-customer and team-oriented vision for space systems design is implemented deliberately and conspicuously. This design setting does not obviate the need to make critical design trades, but it creates the opportunity to discuss and resolve these issues in real time. Not only does this reduce the time and cost necessary in the conceptual design phase (Stagney 2003), but it also provides a closed setting in which the interplay among the various subsystems and disciplines described above can be rigorously and comprehensively analyzed. For this reason, this design environment is used as the subject of data collection for the research presented in this thesis.

3.2. The Purpose and History of Integrated Concurrent Engineering

In this section, a relatively new approach to space systems design called Integrated Concurrent Engineering is introduced. First, the need for integration in the practice of concurrent engineering is explained. Then, the definition and general structure of the ICE environment are discussed. After that, the role of ICE as a tool for lean engineering is described. Finally, the history of the conception and implementation of ICE laboratories in various settings is briefly reviewed.

3.2.1. The Need for Integration in Concurrent Engineering

The traditional approach to developing large-scale engineered systems follows the model of sequential engineering, in which the work of various departments is done separately and in series. Chelsom (1994) presents two case studies that demonstrate the problems that can arise when implementing this "over the wall" approach to product development. In both cases, conducting the program in this way led to schedule delays and significant extra costs. In contrast to sequential engineering, concurrent engineering (CE) is an approach to product development that emphasizes the entire product life cycle from conception to disposal. In concurrent engineering, all parts of the system are considered simultaneously throughout the design process, and an integrated information system is normally put in place to facilitate the necessary coordination and collaboration. Prasad (1996) defines the eight fundamental principles of concurrent engineering: "Early Problem Discovery, Early Decision Making, Work Structuring, Teamwork Affinity, Knowledge Leveraging, Common Understanding, Ownership, and Constancy of Purpose" (p. 170). In addition, he defines the 7 Ts that affect the implementation of CE: talents, tasks, teams, techniques, technology, time, and tools. Successful implementation of CE requires close attention to all of these features.

Although concurrent engineering offers important advantages, it also comes with potential problems. These issues can arise when concurrency is implemented in an established process in which the iterative nature of CE is no longer necessary, making a sequential approach the more appropriate choice. In these cases, CE can lead to wasted effort, cost increases due to unnecessary iterations, or a build-up of errors due to decreased slack time in the project. In addition, the sharing and use of "immature or imperfect information" can lead to a situation that has been called "concurrent chaos" (Prasad 1996, p. 211). Each of these possible problems with concurrent engineering is associated with a lack of integration across those aspects of the design work that are implemented concurrently. McCord and Eppinger (1993) have termed this issue the integration problem in concurrent engineering. Establishing the appropriate means for integrating the various components of a project is a critical part of implementing CE in an organization. Possible mechanisms for integration in concurrent engineering include "direct contact, co-location, liaison role, cross-functional teams, secondment, role combination, permanent project team or cell, and matrix management" (Pawar 1994, p. 52).

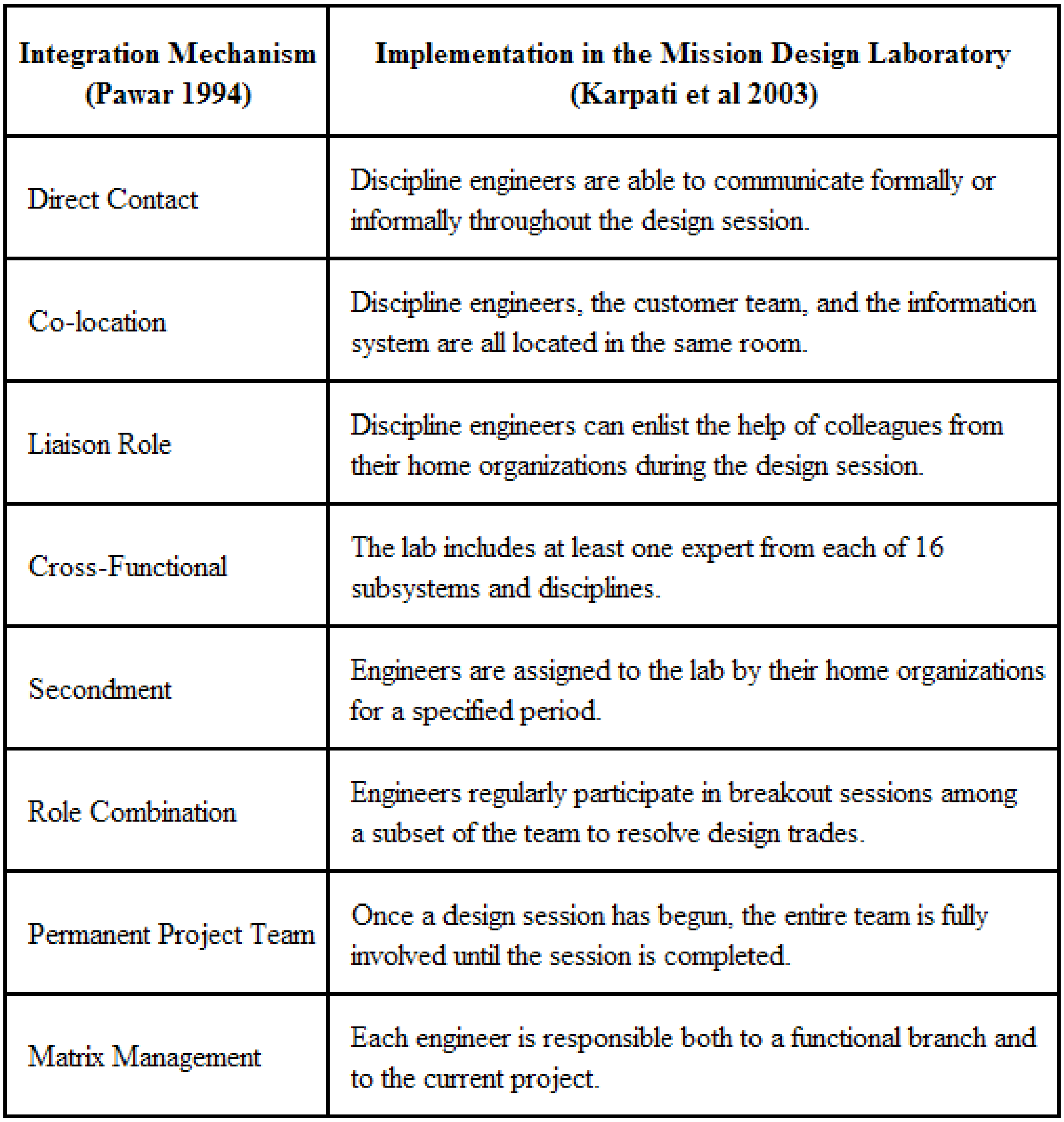
The Integrated Concurrent Engineering design environment implements virtually all of these integration mechanisms. Table 3-1 describes how each of these mechanisms is implemented in a particular ICE design center, the Mission Design Laboratory at NASA GSFC. In this type of design setting, the integration problem is addressed directly and continuously throughout the design process. Because everyone involved is together in the same room, integration is a normal and nearly automatic part of the process. In the next subsection, the general structure of the ICE design environment is described.

3.2.2. What is Integrated Concurrent Engineering?

The purpose of Integrated Concurrent Engineering is to increase the pace of conceptual design by bringing together all relevant personnel to conduct focused, collaborative design sessions within a well-defined timeframe, usually about a week. The ICE environment explicitly removes physical and organizational boundaries to communication so that design tasks that once took months or even years to accomplish can be completed in a matter of days (Sercel et al. 1998, Karpati et al. 2003). These design settings are not only venues for concurrent engineering, but they also are integrated in the sense that the various discipline engineers (usually one per discipline) are collocated in the same room so that they are able to concentrate their efforts on the truly interdisciplinary aspects of the design. For this reason, ICE design teams are characterized by a high degree of expertise and functional diversity. In fact, they are "expert teams" according to the definition offered by Salas et al. (2006) and discussed in Chapter 2 of this thesis.

An ICE laboratory generally includes several work stations that correspond to the subsystems, disciplines, or other necessary functions in the design process. Each work station normally is staffed by one engineer, but there could be two or three people working on certain

Table 3-1. Integration Mechanisms in the Mission Design Laboratory.



Integration Mechanism (Pawar 1994)	Implementation in the Mission Design Laboratory (Karpati et al 2003)
Direct Contact	Discipline engineers are able to communicate formally or informally throughout the design session.
Co-location	Discipline engineers, the customer team, and the information system are all located in the same room.
Liaison Role	Discipline engineers can enlist the help of colleagues from their home organizations during the design session.
Cross-Functional	The lab includes at least one expert from each of 16 subsystems and disciplines.
Secondment	Engineers are assigned to the lab by their home organizations for a specified period.
Role Combination	Engineers regularly participate in breakout sessions among a subset of the team to resolve design trades.
Permanent Project Team	Once a design session has begun, the entire team is fully involved until the session is completed.
Matrix Management	Each engineer is responsible both to a functional branch and to the current project.

disciplines that are particularly important for a given design session. A team lead or facilitator is responsible for the overall progress of the session, and a systems engineer usually leads the technical integration of the design work. A customer team, generally consisting of scientists, systems engineers, and/or program managers, commissions the study and is usually involved either directly or indirectly over the course of the design work.

Since the 1990s, Integrated Concurrent Engineering has become an increasingly recognized and utilized approach to the design of complex systems, especially in space mission design. The growing popularity of ICE can undoubtedly be attributed to its ability to produce a full conceptual design with minimal investment of time, money, and resources. In essence, the ICE environment is a setting for lean engineering. Although lean engineering was not an explicit consideration in the conception of ICE, the two approaches arose from similar sets of needs in different contexts. The purpose of the next subsection is to briefly describe the concept of lean and to discuss the role of ICE as a tool for engineering according to the basic principles of lean.

3.2.3. ICE as a Tool for Lean Engineering

Developed primarily in the automobile industry and popularized by Womack et al. (1990) in *The Machine That Changed the World*, lean is an approach to production that focuses on eliminating waste and creating value for the customer. Rather than a step-by-step set of procedures, lean is really a way of thinking about achieving the goals of an enterprise.

> *Lean thinking is the dynamic, knowledge-driven, and customer-focused process through which all people in a defined enterprise continuously eliminate waste with the goal of creating value (Murman et al. 2002).*

This definition leads to the identification of *seven wastes*: Overproduction, Inventory, Movement, Waiting time, Processing, Rework, and Transportation. Although these categories were defined in the context of manufacturing, they can also be applied to other areas, including design (Murman et al. 2002).

Lean engineering (that is, the application of lean thinking to engineering design) has three basic goals (McManus et al. 2005). The first is to develop the “right products” (p. 2). In the ICE environment, this is accomplished by having the customer directly specify the design requirements to the team at the beginning of each session and by involving the customer in the entire design process. The second goal is to include “effective lifecycle and enterprise integration” (p. 2). This is done in the ICE environment by involving all necessary disciplines and considering all phases of the system life cycle. Finally, the third goal is to implement lean principles to eliminate the seven wastes as they apply in an engineering design context. McManus (2005) offers a list of these so-called “info-wastes,” and Coffee (2006) discusses their application in the ICE environment. The collocation of the ICE team and the rapid pace of the process are intended to eliminate just these types of wastes.

Because the ICE environment is set up to achieve the goals of lean, it can be viewed as an important tool to enable lean engineering (McManus et al. 2005). One of the ways in which this is accomplished is by facilitating "seamless information flow" (p. 4) throughout the process. Thus, the discussion of team coordination presented in Chapter 6 of this thesis directly incorporates lean principles into the analysis.

3.2.4. A Brief History of ICE

The implementation of Integrated Concurrent Engineering began in 1994 at the Jet Propulsion Laboratory (JPL) with the creation of the Product Design Center (PDC) and its design team, Team X (Wall 1999). Stagney (2003) notes that while the effort of Team X to do "true real-time concurrent engineering" (p. 40) was not a new concept at the time, the idea of meeting in the same room to actually do the work together was an innovative idea. Since then, however, several other organizations in government and industry have begun to implement the concept. In 1996, another ICE design center was conceived at NASA Goddard Space Flight Center. The resulting Integrated Mission Design Center (IMDC) began its operations in June 1997. Whereas Team X projects are divided into three-hour sessions booked separately by the customer team (Smith 1998), design sessions in the IMDC normally involve a full design team working together in the facility throughout the entire design study, which usually lasts about a week.

Shortly after the founding of these two design centers, a partner design lab was created for each of them. The focus of the design work done by Team X and the IMDC is on spacecraft and the surrounding mission architectures (generally planetary missions for Team X and Earth-orbiting science missions for the IMDC). The partner labs, on the other hand, were created to design scientific instruments. The partner facility to Team X is known as Team I (Smith 1998), and the IMDC's partner was named the Instrument Synthesis and Analysis Laboratory (ISAL). The IMDC and the ISAL were created as part of an organization called the Integrated Design Capability. In 2007, the Integrated Design Capability was renamed the Integrated Design Center (IDC), and the IMDC and the ISAL became the Mission Design Laboratory (MDL) and the Instrument Design Laboratory (IDL), respectively.

Following on the heels of JPL and GSFC, NASA Langley Research Center (LaRC) began the development of its own ICE facility in 2002 (Gough et al. 2005). The ICE facility at LaRC is known as the Integrated Design Center (IDC). Like the facilities at JPL and GSFC, this design center also has a partner lab, but its function is entirely different. The Mission Simulation Lab (MiSL) is a virtual reality environment that is used "to create a simulation-to-flight capability for LaRC spaceflight projects" (Gough et al. 2005, p. 2). The IDC and MiSL together form an organization known as the Interactive Design and Simulation Center (IDSC).

In addition to these NASA-affiliated ICE laboratories, the European Space Agency (ESA) also operates an ICE design center. In late 1998, ESA established the Concurrent Design Facility (CDF) at the European Space Research and Technology Centre (ESTEC) in Noordwijk, The Netherlands (Bandecchi et al. 2000). In addition, a few ICE design centers have been established in the private sector. For example, The Aerospace Corporation's Concept Design Center (CDC) and TRW's Integrated Concept Development Facility (ICDF) both opened in the late 1990s (Aguilar and Dawdy 2000, Heim et al. 1999). Boeing Satellite Systems (BSS) opened the Concurrent Integrated Engineering Lab (CIEL) in 2002 (Sanders 2002), though the center is no longer in regular operation.

The data collection for this thesis primarily took place at the NASA GSFC Mission Design Laboratory. Rather than conducting a broad-based survey of all ICE design centers, the research focuses on this one ICE facility to allow for greater depth of analysis. Because the effort was dedicated to this one center, the MDL management made the author an official member of the organization, which allowed virtually unlimited access to the facility, the personnel, and much of the information system. Based on this experience, the next section provides a detailed discussion of the MDL.

3.3. The Mission Design Laboratory

As stated above, the ICE design center on which this research is based is the Mission Design Laboratory at NASA Goddard Space Flight Center. The MDL was the best choice for this research for two reasons. First, the MDL normally designs Earth-orbiting spacecraft but has recently begun to take on planetary missions and certain advanced concepts. Thus, this design center offers a broad space of types of design sessions on which to base the analysis. Second, unlike many other ICE laboratories, the members of the customer team are represented as a stakeholder group since they are actively involved throughout the design process. The purpose of this section is to give the appropriate background to understand this particular design environment. The first subsection provides an overview of the organizational context in which the MDL operates. Then, the second describes the MDL in terms of the four elements of its structure – People, Process, Tools, and Facility.

3.3.1. Organizational Context of the Mission Design Laboratory

The organizational context in which the Mission Design Laboratory exists is relevant to the analysis of the design process. As discussed in the previous section, the MDL does not operate alone but rather works closely with a partner lab called the Instrument Design Laboratory. The MDL and the IDL together operate under the Integrated Design Center, which is part of the Systems Engineering Services and Advanced Concepts (SESAC) branch. The operations manager of the IDC is responsible for filling the calendar with design sessions for both the MDL and the IDL. Each laboratory conducts approximately two one-week design sessions per month on average. During periods of heavy workload, this rate can increase. Other times, several weeks can pass without a scheduled design session. On occasion, the MDL also conducts shorter design sessions focused on a small subset of the relevant disciplines. Other times, larger-scale design sessions are conducted. These sessions are often broken into multi-part studies scheduled for separate weeks.

The MDL and the IDL are closely connected to each other in terms of content as well as organization. Although many design sessions are assigned to the two labs separately, a significant portion of the mission concepts handled by the IDC are analyzed in design sessions conducted in both of its labs. In this case, the normal mode of operation is that the customer team first commissions a study by the Instrument Design Lab, which produces a design concept for the mission payload. Soon after, usually within a couple months, the same customer team returns to bring the same concept into the Mission Design Lab. In the MDL, the spacecraft and associated mission architecture are designed around the payload designed during the IDL session. Although some modifications may be made to the instrument during the MDL session, the work of the two labs is normally conducted entirely separately.

During the course of data collection for this research, the IDC went through a few important organizational changes. First, when the research began, the MDL had recently begun to work on a number of mission concepts that are outside of its normal scope of work. Typically, an MDL design session involves the design of a single scientific spacecraft and mission architecture with an orbital trajectory that remains within the influence of Earth's gravity. For the MDL design team, this type of concept is familiar and the design process somewhat routine. As a result of the new orientation of the design sessions, however, nearly half of the observed sessions involved planetary spacecraft or certain advanced concepts outside of the MDL's traditional "comfort zone." As stated above, this change in the normal operation of the lab is an important reason that the MDL process was the focus of the investigation. As will be seen in later chapters, the resulting variety of mission concept studies included in the data set has proven to be beneficial for the outcomes of the research.

The second major change in the MDL since the research began was its renaming, as mentioned in the previous section. The Integrated Design Capability became the Integrated Design Center to clarify that it is an actual organization with its own management structure and personnel. The changes from the IMDC to the MDL and from the ISAL to the IDL were made to establish a more consistent and transparent naming convention. As a result, the two facilities are both labeled as laboratories instead of one center and one laboratory. The change also simplified the names so that the labs' functions would be more apparent to prospective customers. Along with the change in the names, the position previously called Team Lead in each lab became known as Lab Lead (though the former term is still used throughout this thesis). Officially, these changes have become permanent, but much of the staff and the existing customers continue to use the previous names informally, especially with respect to the acronyms. These name changes are notable, but they do not affect the nature of the work or the process implemented by either lab.

3.3.2. Elements of the Mission Design Laboratory

The structure of the Mission Design Laboratory is based on four key elements: People, Process, Tools, and Facility. These four elements, depicted in Figure 3-1, enable the capabilities offered by an Integrated Concurrent Engineering design environment. In this section, each of these elements is described, and their relationship to the research is briefly discussed.

3.3.2.1. People

A typical MDL design session involves approximately 20 to 25 people that are involved directly in the design process. The session is facilitated by a Team Lead, and the technical leadership of the design work is shared between the Team Lead and a Systems Engineer. The MDL design Team includes 16 subsystem and discipline engineers. The disciplines represented are Attitude Control, Avionics, Communications, Electrical Power, Flight Dynamics, Flight Software, Integration and Test, Launch Vehicles, Mechanical, Mission Operations, Orbital Debris, Parametric Cost, Propulsion, Radiation, Reliability, and Thermal.

In general, an MDL session involves one engineer per discipline. In some cases, however, a second engineer might be staffed to provide additional support to a critical discipline. The disciplines for which two engineers were staffed during at least one of the observed design sessions are Attitude Control, Electrical Power, and Propulsion. The roles of Launch Vehicles and Parametric Cost are usually filled by the same person during an MDL session. This is feasible because of the nature of those two disciplines. The work of Launch Vehicles is done entirely at the beginning of the design session, and the work of Parametric Cost is done at the end of and after the session.

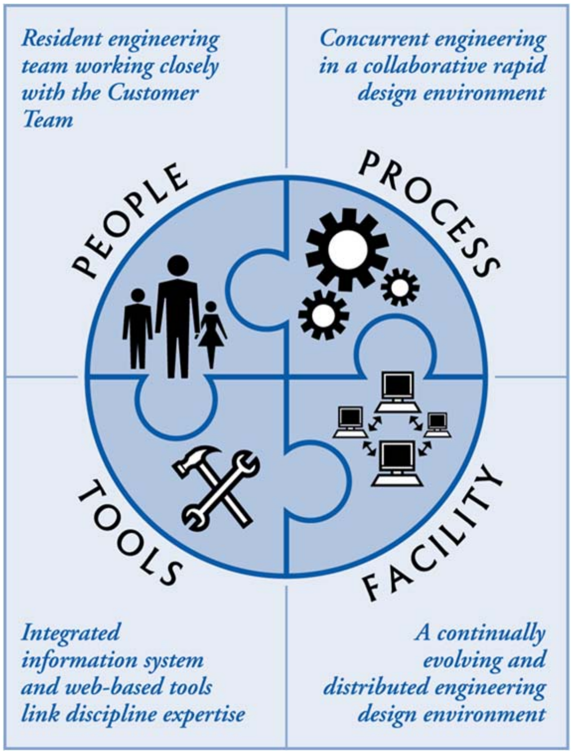


Figure 3-1. The Elements of the Mission Design Laboratory. From Karpati et al. (2003).

As discussed previously, each of the engineers in the MDL holds a full-time appointment with his or her home organization, and the assignment of personnel to the MDL is done by the branch head of each organization. This has important implications for the make-up of the MDL design team from one session to the next. Some branch heads prefer to assign a single expert that can provide dedicated support to the MDL for every session, while other branch heads choose to assign a different engineer to each session based on each person's availability given other professional obligations. For the observed design sessions, the roles that seldom changed staffing were Communications, Electrical Power, Integration and Test, Launch Vehicles and Parametric Cost, Mechanical, Orbital Debris, Radiation, and Thermal. In addition, the Team Lead and Systems Engineer roles are usually filled by the same person from one design session to the next. Each of the remaining disciplines has two or more engineers available, and any one of them could be chosen to participate in a session during a given week.

As mentioned above, the MDL requires full customer participation throughout the entire design session. The customer team can consist of as few as two or as many as eight or more people actively engaged throughout the process. During every design session, the customer team includes at least one of the following members: a Systems Engineer, a Program Manager, and a Project Scientist (or Principal Investigator). For many sessions, two or all three of these roles are filled. During some design sessions, the customer team also includes certain discipline experts. In these cases, each customer discipline engineer generally works side-by-side with the MDL engineer for his or her discipline. The interplay among the people in the room, both among disciplines and between the design team and customer team, enables the integrated and concurrent design process.

The people that form the heart of the Mission Design Lab are also the most essential element of the research presented in this thesis. In general, most parameter dependencies in the design process exist in the form of tacit knowledge, or unspoken know-how of the engineers. The design process model discussed in Chapter 4 was made possible by interviewing the people to determine what information each requires from other members of the team. Furthermore, throughout the data collection period, the members of both the design team and the customer team for each design session completed a pre-session and a post-session survey. The surveys form the basis for the model of shared knowledge proposed in Chapter 5.

3.3.2.2. Process

Broadly speaking, the MDL process includes four steps, which are described in Figure 3-2. Steps 1, 2, and 4 represent the initial customer request, a relatively brief pre-work meeting, and follow-up work, respectively. The activities of the actual design session are captured in step 3. This step is the part that is of primary interest in this research. A typical MDL design session takes five days, usually Monday through Friday of a given week. The design session generally starts at 9:30 am on Monday morning with a briefing from the customer. One or more members of the customer team delivers a presentation` laying out the objectives of the design session and any conclusions that the customer team has already reached. Once this presentation is completed and all questions from the design team answered, the design work begins. Each day, the full team meets for a tag-up meeting at 9:30 am and 1:30 pm. In addition, any small groups of discipline engineers and/or customer team members that need to resolve particular design trades or other issues hold sidebar meetings as needed.

Figure 3-2. The Mission Design Laboratory Process. From Karpati et al. (2003).

During most MDL sessions, the design work is well underway by the time of the first team-wide tag-up meeting at 1:30 pm on Monday. At this point, certain tensions inherent to concurrent design start to arise. For example, during virtually every design session, the Electrical Power engineer makes certain starting assumptions about the power requirements for each other subsystem and then requests that the other engineers provide updated numbers as soon as possible. Until that happens, the Electrical Power subsystem design is of low fidelity.

By Wednesday afternoon or Thursday morning, the team will have completed two to three iterations of the full mission design. The line at which one iteration ends and another begins, however, cannot be neatly drawn because of the large amount of informal interactions among the members of the team. Once this point is reached, the Team Lead and Systems Engineer call for a freeze of the design. Thursday afternoon and Friday morning are normally used by the discipline engineers to create the final presentations that they will deliver at the end of the week. The customer team is generally asked not to attend the Friday morning tag-up meeting. This gives the design team an opportunity to resolve any final issues that could not be settled with the customer team present and to complete their final reports without the possibility of additional requests being made for the design work itself.

At 1:30 pm on Friday afternoon, the final presentation begins. Each discipline engineer presents his or her results to the customer team in turn. After that, most of the design team has completed the work for that session. The Parametric Cost engineer, however, only just begins the cost estimation work at that point.[3] Once the cost estimate has been completed, the Team Lead and Systems Engineer wrap the study and plan a post-work meeting with the customer

---

[3] Although costing is generally an integral part of the process in full-scale development programs, it is completed only at the end of each MDL session. This is discussed further in Chapter 4.

team. This normally takes place a few weeks after the design session has been completed. As a result, the final work of the Team Lead, the Systems Engineer, and the Parametric Cost engineer lags the design session calendar by several weeks.

This well-defined process enables the team to complete studies on a variety of mission concepts in an accelerated timeframe. Not only does this increase the efficiency of the design center, but it also makes the MDL an ideal laboratory in which to collect data for this research. The standardized approach makes it possible to track parameter dependencies and to create a model of information flow that is applicable across all design sessions. The large number of sessions completed in a short time offers the opportunity to collect pre-session and post-session survey data on several design sessions and to observe the sessions in their entirety. The means of data collection will be described in section 3.4, and the analysis of the design process will be presented in Chapters 4, 5, and 6.

3.3.2.3. Tools

The tools used by the MDL design team come in two types: discipline tools and integrated system tools. The discipline tools are generally chosen by each discipline engineer based on their own preferences and familiarity. The tools can be government off-the-shelf, commercial off-the-shelf, or custom-made by the discipline engineers. Although the MDL does not take control of the individual tools chosen, the lab does provide an opportunity for engineers from different disciplines to share tools and thereby expand the capabilities of their home organizations (Karpati et al. 2003).

The integrated system tools used by the MDL to facilitate concurrent engineering have been continuously evolving since the design center was established. Initially, the team used only traditional means like verbal communication and e-mail for data exchange, but the center quickly implemented a tool called the IMDC System for Information Sharing (ISIS). This tool proved to be invaluable for the design of single-spacecraft missions to low Earth orbit (LEO), but it could not be modified as the lab began to take on more complex mission concepts because it was written in static HTML. As a result, the center created a new tool called the EXcel Information eXchange (EXIX), which was more flexible than ISIS because it was written in Visual Basic underneath an Excel front-end (Karpati et al. 2003).

Today, the MDL uses an even more flexible and dynamic tool called the Process Reasoning and Information Management Environment (PRIME). PRIME allows each discipline engineer to upload output parameters from his or her own discipline and to retrieve parameters posted by other discipline engineers as needed. The tool is also useful to the Systems Engineer for bookkeeping and overall tracking of the design. The tool is accessible via a web interface and can also be used to generate Excel-based reports. When necessary, PRIME can be modified to store information for multiple stages or phases of high-complexity mission architectures. Finally, PRIME stores data about all previous design sessions for which the tool was used, allowing the team to quickly retrieve relevant information from those sessions.

The next step in the evolution of the MDL information sharing capability is a tool that will allow real-time system-wide updates. With such a tool, any changes made to one subsystem design could be propagated throughout the entire design at the touch of a button. JPL's Team X already uses a tool called ICEMaker™ that serves just this type of function (Parkin et al. 2003). Developed at the California Institute of Technology's Laboratory for Spacecraft and Mission Design, ICEMaker is built on client-server architecture. The ICEMaker server manages information for the entire design, and each discipline engineer controls one of the clients, reporting changes to his or her design and accessing changes that affect his or her work by querying the server.

The MDL's PRIME tool, on the other hand, does not have the capability to make system-wide updates or to determine how changes in one discipline affect others. In some ways, certain features of the MDL make such a tool less necessary than it is for Team X. First, because the entire team works together in the MDL facility for the entire week rather than just during a few three-hour sessions, most of the relevant design trades and multidisciplinary issues are handled through direct person-to-person communication. Secondly, in contrast to Team X, the customer Team is continuously involved in an MDL design session and thus would want to be privy to any system-wide changes before they happen. Therefore, it is not entirely unreasonable that the MDL has not yet made the investment in an ICEMaker-type tool.

Although PRIME does not have all of the capabilities of ICEMaker, it does enable the engineers on the team to track parameters from most other disciplines as they change. Furthermore, it provides real-time information about routine or uncontroversial changes in the design of individual subsystems and disciplines, which frees time for the Systems Engineer to

Figure 3-3. The Mission Design Laboratory Facility. *Source:* Integrated Design Center, NASA Goddard Space Flight Center.

focus on the most important and/or difficult issues during discussions with the discipline engineers. Moreover, PRIME has provided an important advantage for this research. Not all parameter flow in the MDL design occurs via PRIME, but the tool catalogs most of the important parameters and offers an overview of many of the technical properties of the system being designed. For this reason, PRIME served as the basis for initial survey questions on the passing of parameters among discipline. The parameter data were later refined through structured interviews with each discipline engineer, but the structure of those interviews and thus the completeness of the resulting data was enabled by the information provided in PRIME.

3.3.2.4. Facility

Perhaps the most readily apparent element of the MDL is its facility, shown in Figure 3-3. The layout of the facility, which is shown in schematic form in Figure 3-4, is intended to encourage free exchange of information, both formally and informally, among the discipline engineers and the customer team. The laboratory portion of the facility is approximately 1000 square feet in size and contains about 20 work stations that each corresponds to a role on the design team (Karpati et al. 2003). At the front of the room is a round table around which the

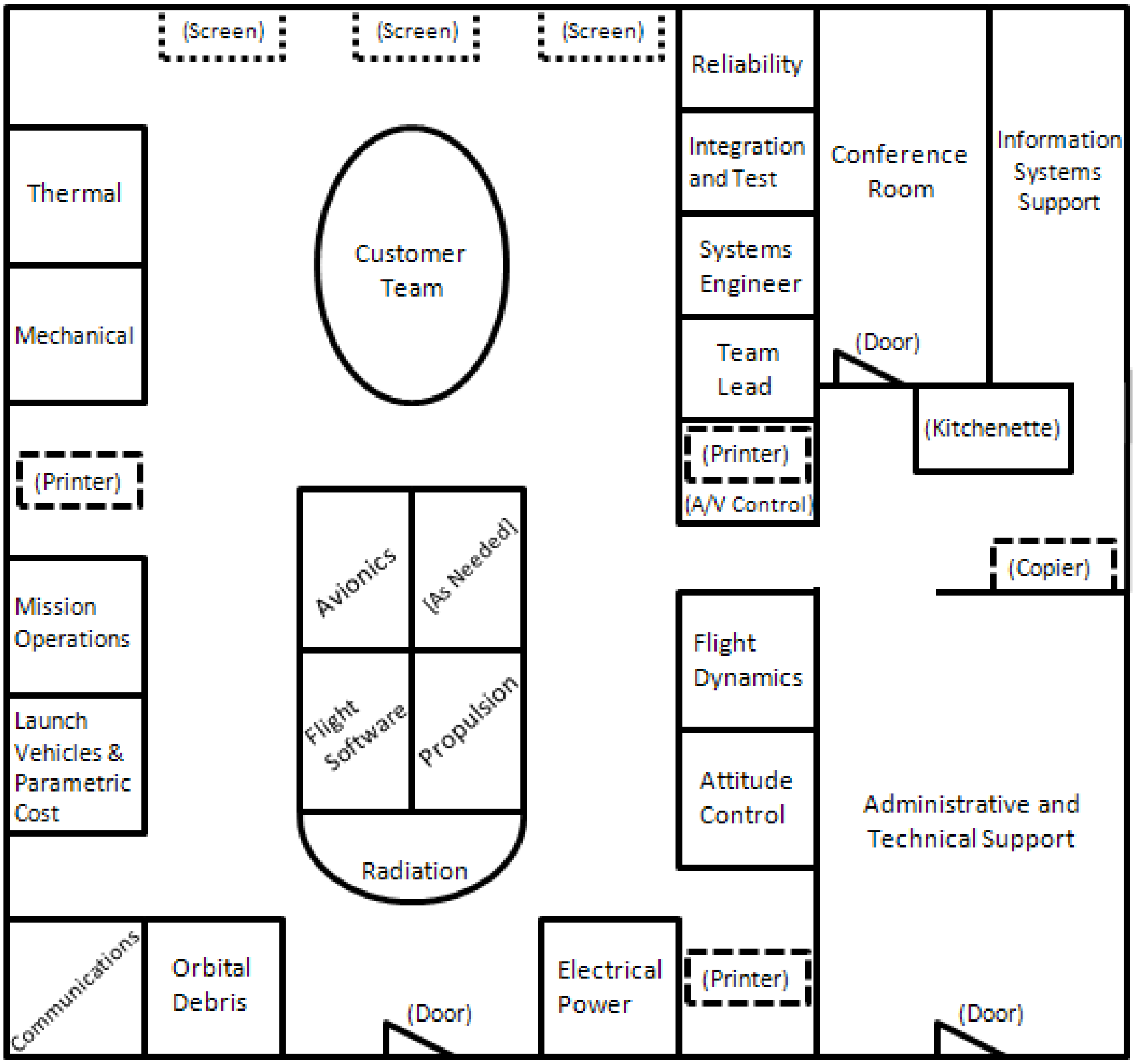


Figure 3-4. Layout of the MDL Facility. The main lab holds work stations for all of the discipline engineers, a table at the front for the customer team, and a full audio-visual system. Next to the main lab are a conference room used for small breakout sessions, the offices of the MDL support staff, and a kitchenette stocked with coffee and snacks.

customer team sits throughout the design session. A small conference room used for breakout sessions is located adjacent to the lab, and a small kitchenette (normally stocked with coffee and snacks) is just outside of the conference room.

In addition, the support staff work space is part of the facility and is directly accessible from the main lab. The information systems support staff is located next to the conference room, and the administrative and technical support staff is across from the kitchenette. Without the efforts of the support staff to ensure that the facility and the tools are running properly, the design team would not be able to conduct their work effectively. Because the members of the support staff are collocated with the design team, they are readily available to resolve any issues that arise with the audio-visual equipment, the workstations, the network, or any another component of the MDL infrastructure.

Each work station in the lab runs the latest Windows operating system and can be accessed by any member of the MDL team using their GSFC username and password. In addition, the MDL information system and the common files containing products of past sessions are available from each work station. The facility is furnished with a full audio-visual system, including three ceiling-mounted projectors and three projector screens on the front wall. The room also includes a wireless microphone, which the Team Lead uses to run tag-up meetings and discipline engineers use to provide updates to the entire team. Three network printers are available within the main lab and can be used from any of the work stations. In addition, a copier is available near the information systems support area.

The arrangement of the work stations in the lab has evolved since the MDL was established. Many of the seating assignments are made because certain disciplines are expected to interact frequently and thus should be close to each other. For example, Flight Dynamics, Attitude Control, and Propulsion all deal with issues relates to the dynamics of the spacecraft. Similarly, various subsets of Flight Software, Avionics, Mission Operations, and Communications work together to resolve trades regarding storage, processing, and transfer of data. Furthermore, the Team Lead and Systems Engineer obviously should be seated next to each other and near the customer team since they are responsible for managing the project.

In other cases, however, certain disciplines are seated on a space-available basis. For example, there is no particular reason for Orbital Debris to be located near Communications and Flight Software. Reliability was recently moved to the front corner of the room because the previous location near Communications was simply getting too crowded. In addition, certain pairs of closely connected disciplines, such as Flight Dynamics and Communications, are not located near each other at all. In fact, Electrical Power and Thermal are highly interdependent and yet are located at opposite corners of the room. Therefore, it is a common occurrence during an MDL session to hear the Electrical Power engineer call across the room to ask a question of or to provide information to the Thermal engineer.

Even when closely connected disciplines are located “far” from each other in the room, however, they are still located close to each other by most standards and are able to interact whenever necessary. Thus, the MDL facility provides a valuable central location in which virtually all of the collaborative work for each design session takes place. In addition, the MDL facility also provided the essential “laboratory” that made the research presented in this thesis possible. Because all of the relevant discipline engineers are present in the room and actively participate throughout the process, the MDL offers an ideal setting in which to conduct generalizable research on the relationship between the technical design process and shared knowledge in the team. The next section discusses how the MDL facility enabled this research and then discusses the observation-, survey-, and interview-based methods of data collection on MDL design sessions.

3.4. Data Collection in the Mission Design Laboratory

The Mission Design Laboratory provides a “semi-controlled” setting in which to collect data on the relation between the engineering design process and team dynamics. The research is semi-controlled in the sense that all design sessions follow a standardized process with only few specific differences. It is not completely controlled, though, for a couple reasons. First, there is more than one variable that changes between sessions. For example, the precise concept under study and many of the team members change from one session to the next. Still, the process, the tools, the facility, and about half of the team members remain the same. Second, the researcher did not have the ability to actively vary the parameters of the study but rather collected data on the sessions that were being held. Nevertheless, the benefit that comes from doing research in such an environment is that it is a real-world setting, so the insights are based on the work of actual engineering design teams. Therefore, the MDL provides many of the benefits of both a controlled laboratory environment and a real-world design setting, and the results are thus applicable to both theory and practice.

The purpose of this section is to review the specific process of data collection and the type of data collected in the Mission Design Laboratory. The first subsection enumerates, describes, and categorizes the 12 MDL design sessions that were observed over the course of the data collection period. The second subsection focuses on the structure of the surveys that the members of the design team and the customer team completed before and after each of the 12 observed design sessions. Finally, the third subsection explains the format of the interviews used to track information flow in a typical MDL design session.

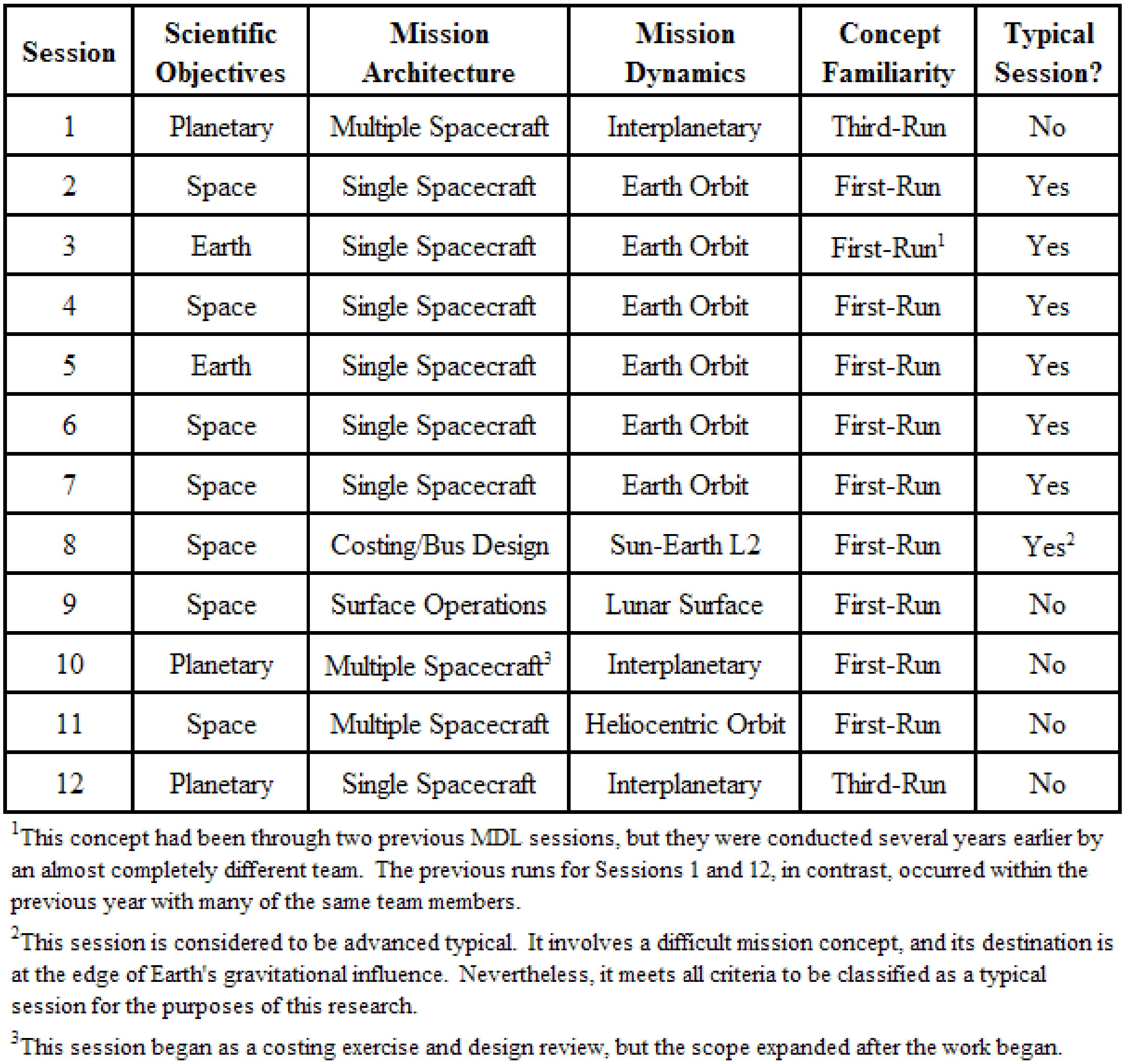
3.4.1. Observations of Design Sessions

The basic component of the data collection was the observation of 12 design sessions over an eight-month period. Although the data used for the formal analysis came from surveys and interviews, the design session observations formed an important part of the data collection process. The observations did not include formal note-keeping or tracking of specific events and conversations, but it enabled the researcher to understand the people, the process, the tools, and the facility so that the results could be interpreted with respect to the features of each design session. From these observations and subsequent reviews of the design products, the researcher characterized each session along several dimensions. The list of all 12 design sessions observed and the classification of each are provided in Table 3-2.

Each design session in the table is classified according to scientific objectives, mission architecture, mission dynamics, concept familiarity, and whether or not it was a typical design session. The scientific objectives are classified as one of three types: Earth, space, or planetary. Among the 12 session, seven were space science missions, two were Earth science missions, and three were planetary missions. The mission architecture can be one of two broad varieties: single- or multiple-spacecraft. In some special cases, however, the architecture departed somewhat from these categories. Among the observed sessions, session 8 involved a single-spacecraft architecture, but it was a particularly complex spacecraft. In that session, the customer team requested only a design for the spacecraft bus and a cost estimate for the program rather than a full mission design. Session 9, on the other hand, involved a completely different type of architecture – surface operations on the Moon. Of the 10 remaining sessions, seven involved a single spacecraft, and three required a multiple-spacecraft configuration.

Mission dynamics refers to the orbit and other gravity-related influences that affect the mission design. Of the 12 mission concepts, six involved Earth-orbiting spacecraft, and three had interplanetary trajectories (the three planetary science missions, of course). Sessions 8 and 11 involved different types of dynamics from standard Earth-orbiting or planetary missions. They were to be located at the Sun-Earth L2 libration point and in an Earth-trailing heliocentric

Table 3-2. MDL Design Sessions Observed in this Research.



Session	Scientific Objectives	Mission Architecture	Mission Dynamics	Concept Familiarity	Typical Session?
1	Planetary	Multiple Spacecraft	Interplanetary	Third-Run	No
2	Space	Single Spacecraft	Earth Orbit	First-Run	Yes
3	Earth	Single Spacecraft	Earth Orbit	First-Run[1]	Yes
4	Space	Single Spacecraft	Earth Orbit	First-Run	Yes
5	Earth	Single Spacecraft	Earth Orbit	First-Run	Yes
6	Space	Single Spacecraft	Earth Orbit	First-Run	Yes
7	Space	Single Spacecraft	Earth Orbit	First-Run	Yes
8	Space	Costing/Bus Design	Sun-Earth L2	First-Run	Yes[2]
9	Space	Surface Operations	Lunar Surface	First-Run	No
10	Planetary	Multiple Spacecraft[3]	Interplanetary	First-Run	No
11	Space	Multiple Spacecraft	Heliocentric Orbit	First-Run	No
12	Planetary	Single Spacecraft	Interplanetary	Third-Run	No

[1] This concept had been through two previous MDL sessions, but they were conducted several years earlier by an almost completely different team. The previous runs for Sessions 1 and 12, in contrast, occurred within the previous year with many of the same team members.

[2] This session is considered to be advanced typical. It involves a difficult mission concept, and its destination is at the edge of Earth's gravitational influence. Nevertheless, it meets all criteria to be classified as a typical session for the purposes of this research.

[3] This session began as a costing exercise and design review, but the scope expanded after the work began.

orbit, respectively. Session 9 involved special circumstances in which the dynamics simply involved landing and then operating on the surface of the Moon, but there was no need for propulsion or attitude control after landing.

Concept familiarity is directly related to the MDL design team's experience with the mission concept and the customer team. The content of the column indicates the number of times that the same concept has been through an MDL design session. This is not based simply on whether the concept had been evaluated previously within the facility. Instead, it is based on the team members' overall experience with the mission concept during MDL sessions in which they personally participated. Sessions 1 and 12 are both labeled as third-run design sessions. This means that a significant portion of the team had worked on those mission concepts in two prior sessions. In both cases, those two design sessions had included most of the same team members. Thus, the team as a whole worked on those concepts for the third time during sessions 1 and 12. Session 3 was also studied in two prior MDL design sessions, but those previous sessions took place several years earlier. This is important for two reasons. First, it means that most of the design team (except for a couple of people) had not worked on the concept previously because the personnel of the MDL changes over time. Second, because so much time had passed, even those team members that had seen the concept prior to the session were unlikely to remember as much about it as they would have for sessions 1 and 12. For these reasons, the work was effectively new to the team when session 3 was conducted. Therefore, this session is classified as first-run for the purposes of this research.

The last attribute by which the design sessions are classified is essentially an aggregate of scientific objectives, mission architecture, and mission dynamics. Since the MDL began its operations, the typical type of mission concept studied in the MDL has been an Earth- or space-science mission involving a single spacecraft intended to operate within the influence of Earth's gravity. Based on these criteria, sessions 2 to 7 are all classified as typical MDL design sessions, and sessions 1, 9, 10, 11, and 12 are not. For session 8, the classification is less clear than for the others. Although the objective was space science and the architecture single-spacecraft, the session was focused on a major program and a particularly complex spacecraft. In addition, the mission dynamics placed the spacecraft, by definition, at the edge of the Earth's gravitational influence. For this reason, the session is classified as "advanced typical." Technically, it meets all of the criteria for a typical mission. Furthermore, because the session included only a bus design and cost estimate, the work was somewhat less complex than it would have been for a full mission design. Therefore, session 8 is considered in this research to be a typical design session and is classified accordingly in Table 3-2.

Note that among the 12 observed sessions, only seven are classified as typical. The reason for this is simply that the MDL, by coincidence, conducted more atypical design sessions than normal during the data collection period. This has proven to be quite beneficial for the research because it resulted in sufficient data to analyze differences in the dynamics of the team for typical versus atypical sessions. The next section provides an overview of the surveys from which these data were collected.

3.4.2. Design Team and Customer Team Surveys

For each of the 12 MDL sessions, the design team and the customer team members were all asked to complete both a pre-session and a post-session survey. A sample of each of these surveys is provided in Appendix A. Among both the design team and the customer team, the median total number of respondents to both the pre-session and the post-session survey was 20. These surveys usually came from the 16 discipline engineers, the Team Lead, the Systems Engineer, and 2 to 4 members of the customer team. The response rate from the design team each week was 100% except during two sessions.[4] In session 5, the Avionics engineer did not complete the post-session survey, and in session 12, the Systems Engineer did not complete the post-session survey. The size of the customer team ranged from 2 to 8 or more for each session. For the larger customer teams, however, many of the members were not actively engaged during the session. Normally, the customer responses were provided by members of the team that were among the most active in the lab throughout the design session. Therefore, the customer response rate was sufficient for all design sessions observed.

The most important survey question used to assess shared mental models in the team was based on the participants' perceptions of the major design drivers for the session. Because of the MDL's high level of customer participation in the design sessions, this question was asked of the customer team as well as the design team. In addition, this question was asked on both the pre-session and the post-session surveys to provide data on the dynamic nature of shared knowledge over the course of the session.

The question on major design drivers made up the core of the pre-session surveys, but the post-session surveys included a few additional questions whose purpose was to determine the maturity of the concept under study, the team's communication patterns over the course of the session, and technical information flow in the design. Each member of both the customer team and the design team was asked to provide a measure of the technological maturity of the entire

---

[4] In a few cases, two engineers staffed a given discipline. Although responses were collected from both people in many of these instances, a response is considered to have been collected for a given discipline if at least one engineer staffing it responded. In addition, the data for session 5 also includes one member of the design center management, who was actively engaged in the work for that design session.

mission concept, and the subsystem and discipline engineers were asked to make the same assessment for the technology involved in their own design work when relevant. In addition, both teams were asked to provide their assessment of the importance of interactions with each of the other members of the team. Finally, the members of the design team were asked to indicate the parameters that they took as inputs to their work from each of the other subsystems and disciplines. The parameters were listed in sections according to the subsystems/disciplines that provide them, and a checkbox was given next to each parameter. These data provided a coarse-grained data set describing technical information flow in each design session. It did not, however, include the specific dependencies of one parameter on another. These survey data were used as a baseline to guide structured interviews on detailed parameter flow among the subsystems and disciplines. The next subsection provides a description of the interview process.

3.4.3. Design Process Interviews

One popular definition of a "complex system" is one in which there are so many components and processes that it is not possible for any one person to fully understand the entire system. Instead, many people each understand their own parts of the system and the interfaces with other parts. Therefore, any representation of an entire system requires information from everyone involved. This, however, is not as simple as reviewing documentation because much of the information exists in the minds of the individuals. Thus, the only way to obtain all of the necessary information is by conducting interviews with each person involved in the design and/or development of the system.

One effective tool for creating a top-level view of a complex system is the Design Structure Matrix (DSM). As discussed in Chapter 2, the DSM maps out the dependencies among tasks, components, parameters, organizations, or people involved in a system. In this research, a series of interviews with MDL discipline engineers were used to construct a parameter-based DSM documenting information flow in the design process. The process of collecting data on the MDL design process included four steps. First, a baseline list of parameters for each subsystem and discipline was obtained from PRIME. Second, survey data on parameter dependencies were collected after each of the 12 observed sessions as discussed in the previous subsection. Third, a series of structured interviews was conducted with one or more engineer representing each subsystem or discipline. These interviews formed the main part of the data collection on information flow in the design process and were highly iterative due to the interdependence of responses. For example, if a response from engineer B affected an answer provided previously by engineer A, it became necessary to revisit certain issues with engineer A before completing the process. The fourth and final step of the data collection was a verification phase in which each discipline engineer commented on a flow-graph representation of inputs to their own work. This step also included a review of the entire DSM with the MDL Team Lead, who offered a systems-level view that clarified any outstanding issues or conflicting responses given by the discipline engineers.

Because of the nature of the ICE environment, the DSM interviews differed slightly from those normally conducted to build DSM representations of other systems. Specifically, a set of three guiding principles was adopted to account for the ubiquitous information flow and rapid pace of work in this type of design setting. In addition, since only a single DSM was constructed for the general MDL process, the modeled information flow is based on a *typical* design session as defined in section 3.4.1. In Chapter 4, the process of DSM construction in the ICE environment is described in greater detail, and the insights gained from analyzing the MDL process are presented.

3.5. Data Analysis in the Mission Design Laboratory

This chapter has offered a description of the Integrated Concurrent Engineering design environment and the process of data collection in that setting. In the next three chapters, the analysis of the data is presented. Chapter 4 explains the DSM-based representation of the design process, and Chapter 5 proposes a network-based methodology for analyzing the survey data on shared knowledge in the design team. Finally, Chapter 6 integrates the results of the two previous chapters and provides several insights that can be gained from an interdisciplinary analysis of a semi-controlled but real-world design setting like the ICE environment. Based on the results presented in those three chapters, the remainder of the thesis then offers conclusions, recommendations, and opportunities for future work in the ICE environment and in space systems design and development in general.

Chapter 4

The Design Structure Matrix: An Analysis of the Space Mission Design Process

As discussed in the previous chapter, the Integrated Concurrent Engineering (ICE) design environment is fast-paced and highly collaborative setting in which everyone talks to everyone else throughout the entire process. Although some team interaction occurs during scheduled tag-up meetings and periodic breakouts among subsets of the team, much of the information flow in the design process occurs informally. While this aspect of the environment creates opportunities to resolve issues quickly, it also makes the task of tracking information flow a daunting one. Still, it can be done if certain principles are adopted to guide the type and nature of information flow that is actually tracked.

This chapter introduces a method for tracking parameters in the ICE design process and presents an analysis of information flow. Section 4.1 provides an overview of the Design Structure Matrix (DSM) and explains its advantages for design process analysis. Section 4.2 describes how the DSM methodology can be applied to the particular type of process employed in the ICE environment. Then, Section 4.3 presents a DSM-based analysis of the ICE process that reveals the structure and phases of the design life cycle. Next, section 4.4 proposes a technique for analyzing the loops in the DSM to identify critical design trades and interdependent disciplines in the team. After that, section 4.5 explores the effect of making certain starting assumptions at the outset of the work. Finally, section 4.6 discusses some implications of the DSM for the ICE environment and for space systems design in general.

4.1. Overview of the Design Structure Matrix

The Design Structure Matrix is a means of representing an entire system, product, or process by aggregating individual interactions among entities (Browning 2001). It is essentially an N^2 diagram like those often used to manage space systems design, but it is structured in such a

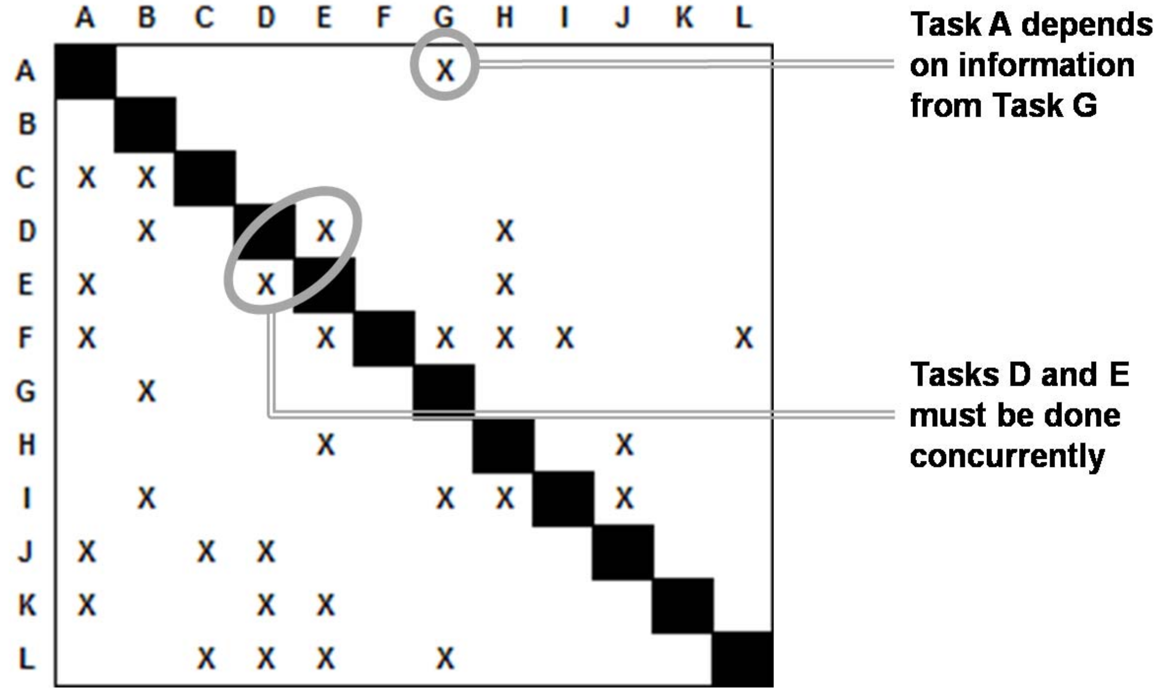


Figure 4-1. **The Design Structure Matrix.** In the DSM, a mark indicates that the task in the row depends on information from the task in the column.

way as to facilitate systems level analysis and process improvement. Each row and corresponding column in the matrix represents a single task or component, and the cells of the matrix indicate dependencies in the process. For this reason, the DSM is also known as the Dependency Structure Matrix. In the matrix, if the task in row *i* requires the task in column *j* as an input, a mark is placed in cell *i,j* (Eppinger et al. 1994). If the type or extent of the dependency is important, a specific kind of mark or a number might be used. Otherwise, a “1” or an “X” is sufficient to denote a dependency in the process. If tasks *i* and *j* depend directly on each other, a mark is placed in both cells *i,j* and *j,i*, indicating that the two tasks must be completed concurrently. Figure 4-1 provides an example of a DSM with marks indicating the dependencies among tasks. For the mathematically initiated, Appendix B describes the mathematical formalism of the DSM. In that discussion, the dependencies among inputs and outputs are explained in terms of function notation.

As mentioned in Chapter 2, a DSM can come in one of four forms depending on the type of dependencies that are represented: component-based, team-based, activity-based, or parameter-based (Browning 2001). The analysis of the space mission design process presented in this chapter is done using a parameter-based DSM, in which the dependencies represent inputs and outputs among parameters. In addition, a technique is proposed for converting the parameter-based DSM into a team-based DSM. In a team-based DSM, the rows and columns denote entities (departments, teams, or individuals) in an organization, and the dependencies indicate those entities that must work together to accomplish the organization's goals. In the ICE environment, the organization is the design team, and each entity is generally an individual team member representing a subsystem or discipline involved in the process.

The DSM is a powerful tool for design process analysis because it combines some of the most important advantages offered by the Gantt chart, Program Evaluation and Review Technique (PERT), the Structured Analysis and Design Technique (SADT), and Quality Function Deployment (QFD). The DSM represents the sequence of tasks in the project timeline in a similar way to the Gantt chart (though the DSM does not indicate the exact timing of the tasks). It also depicts dependencies among parameters in a format that shows the order in which tasks are to be executed just as the PERT chart does. Like SADT, the DSM captures detailed information flow, including feedback and rework. In fact, a SADT diagram and a DSM depict the same information flow – the former as a network graph and the latter as a matrix. Thus, a SADT document can be converted into a matrix for DSM-based analysis (Eppinger et al. 1992, Eppinger et al. 1994).[5] Like QFD, the DSM's matrix format provides a simpler representation than does SADT, but the DSM also improves on QFD for use in process analysis because it represents directional flow of information. Whereas QFD describes dependencies as non-directional correlations in a triangular half-matrix, the DSM uses the entire matrix so that dependencies between pairs of tasks can be depicted in either direction. Therefore, the DSM combines SADT's depth of detail with QFD's simplicity and accessibility while maintaining the relative timing and task sequencing that Gantt and PERT charts provide.

In addition to combining the advantages of the traditional system representation techniques, the DSM also facilitates system-wide process analysis based on an aggregation of information flow. The DSM comes with a toolbox of analysis procedures that can be used to extract new systems-level insights and contribute to process improvement. Specifically, this toolbox consists of three analysis procedures called partitioning, tearing, and clustering. The first two of these

---

[5] The distinctions among inputs, controls, and mechanisms used in SADT can also be made in the DSM by marking each cell of the matrix in a different way according to the type and/or strength of each dependency. In this thesis, however, all types and strengths of dependencies are taken to be equivalent.

techniques can optimize the design process in a task- or parameter-based DSM, whereas the last is generally used to determine logical groupings of entities in a component- or team-based DSM (Gebala and Eppinger 1991). These analysis procedures will be explained as they are used later in the chapter. The next section describes the application of the DSM to the ICE environment.

4.2. Building a DSM for the ICE Environment

This section discusses the process of constructing a DSM in the ICE environment. First, the applicability of this methodology to ICE is demonstrated. Then, the specialized procedure used to create the DSM is described. Finally, the resulting DSM and some of its important features are presented.

4.2.1. The Applicability of the DSM to ICE

The DSM and the ICE environment were both created to cope with the inherent complexity in the design of engineered systems. Despite this shared goal, however, the two approaches have not previously been employed together. The lack of attention to ICE in the DSM literature (and, conversely, the lack of application of the DSM by ICE practitioners) could be a result of a number of factors. ICE and the DSM handle complexity management in different ways. Whereas the goal of the DSM is to identify tasks that are inherently coupled and to decouple those that are not, the purpose of ICE is to foster continuous communication so that tasks do not need to be decoupled at all. Given this difference, it might at first seem counterintuitive to apply these two approaches to the same project.

In addition, the DSM and ICE were conceived in different contexts to handle complexity at different levels. The DSM is normally applied to the detailed design and development of relatively complex products such as automobile parts and aircraft engines. ICE, on the other hand, is usually employed in the early conceptual design phase of even more complex systems – entire spacecraft and the surrounding mission architectures. Indeed, the process of constructing a DSM is a time-consuming and resource-intensive task in itself. Although its use helps to manage complexity, the upfront investment required to initially build a DSM might simply be prohibitive beyond a certain level of system complexity. Moreover, each ICE project is usually completed in approximately one week, so it would not be worthwhile to invest in the construction of an entirely new DSM for every design session.

The ICE environment, however, lends itself to the use of the DSM for two important reasons. First, the ICE design process focuses on interactions among disciplines and subsystem engineers, and the DSM is intended to analyze precisely these types of interactions among the various parts of a system. Second, the ICE environment is characterized by constant information flow and tightly coupled tasks, and one of the most important advantages of the DSM is its ability to identify the sequence of information flow and the coupling of tasks in the process. Still, constructing a DSM for the ICE environment requires a specialized procedure to account for particular features of ICE. It is to this procedure that the discussion now turns.

4.2.2. DSM Construction on ICE: A Specialized Procedure

In general, the construction of a Design Structure Matrix is itself an iterative process. In the Mission Design Laboratory (MDL), the process began with a list of most of the important parameters in the ICE process, which was obtained from the MDL's data exchange tool, the Process Reasoning and Information Management Environment (PRIME). This list of parameters was used as the basis for a series of online surveys. The surveys were distributed to the design team following each of the 12 design sessions observed. The resulting survey data included the specific parameters that each team member requires from other disciplines to complete his or her work. These coarse-grained dependencies were then used as the basis for structured interviews intended to track the passing of specific parameters from one team member to another. A DSM representation of parameter flow was then created from the interview data. Once the DSM was constructed, the final step was an iterative verification procedure using a discipline-centric flow graph representation of inputs for each discipline. In this step, each engineer commented on the graphical representation of his or her own work, and the Team Lead provided a systems-level perspective to the verification.

The steps for DSM construction in the ICE laboratory are summarized as follows:

1) *Review of Existing Documentation*
2) *Surveys on Design Sessions*
3) *Structured Interviews*
4) *Model Verification*.

These four steps are similar to the general procedure for constructing a DSM on any project or system. The specific questions asked during each phase, however, must be modified in the ICE environment. A few features peculiar to the rapid design setting directly affect the nature of the data needed for DSM construction. Because of these peculiarities, three guiding principles have been adopted in creating the DSM. Each of the principles follows directly from one of the characteristics of ICE. The characteristics, their implications for DSM construction, and the guiding principle that follows from each are enumerated in Table 4-1.

The first characteristic of the ICE environment is the standardized process that is used to accommodate a large number of design sessions. This standardization makes building a separate DSM for each design session both unnecessary and excessively resource-intensive. Because of the frequency and short duration of the sessions, the standard ICE process is modified only when needed from one session to the next. Therefore, a generic DSM representation of a typical design session includes a large portion of the information flow in most sessions. In the Mission Design Laboratory, for example, a typical session involves the design of a single Earth-orbiting spacecraft for Earth or space science and the associated mission architecture. To ensure that the generic DSM contains the information required for all typical sessions, it must include the maximal flow for such a session. In this context, maximal flow refers to all information that is actually passed or that at least must be considered, even if the value of a given parameter is "not applicable" in some sessions. Although some of these parameters might not be needed by some disciplines in a particular session, all possible flows are included in the DSM since the design team will not know in all cases whether a piece of information must be passed along until they reach the relevant point in the design.

The second characteristic of the ICE environment is that the team is collocated in the design facility for most or all of the process. Therefore, everyone talks to everyone else throughout the session. This occurs to some extent during the regularly scheduled tag-up meetings, but the bulk of this communication occurs through informal interactions over the course of the work. This ubiquitous information flow unquestionably improves design outcomes and, in fact, is what makes the ICE approach particularly valuable. Still, the benefit of these unstructured interactions is serendipitous and does not represent "typical" information flow. Furthermore, including all such interactions in the DSM could result in a matrix that is so densely populated that it would convey little useful insight. Therefore, the second guiding

Table 4-1. Guiding Principles for DSM Construction in the ICE Environment. Each of the characteristics of ICE has certain implications for DSM construction that lead to one of the guiding principles.

	Characteristic of the ICE Environment	Implications for DSM Construction on ICE	Guiding Principle for DSM Construction on ICE
1	Standardized process for a large number of design sessions	A separate DSM for each session would be both unnecessary and excessively resource-intensive.	Document maximal information flow for a typical design session
2	Frequent communication resulting from collocation of all personnel in a single facility	Some interactions in the ICE setting are extraneous and would make the DSM less representative of the typical process.	Include only deliberate and purposeful information flow
3	Ubiquitous two-way negotiations between discipline engineers on important parameter trades	Most communication is two-way, so documenting these interactions in both directions would overcomplicate the DSM without providing additional insight.	Abstract two-way negotiation-type interactions to show only net flow of information

principle employed is that only deliberate and purposeful information flow is included in the representation.

The third characteristic of the ICE environment is a direct result of the second. Because of the continuous and open communication, a kind of negotiation between pairs of parameters occurs on a regular basis. This negotiation is much less common in sequential engineering because it simply is not practical in that context. To understand the negotiation, consider a trade between two parameters from different disciplines. In many ways, the trade is analogous to a marketplace negotiation. Two agents haggle over the price and the conditions of the purchase, but when the negotiation is complete, there is a clear buyer and a clear seller involved in the transaction. Similarly, in the ICE design process, two discipline engineers might negotiate certain pair-wise design issues, but when the discussion is complete, one generally uses the information, albeit in a modified form, from the other. Since this occurs so frequently in an ICE laboratory, documenting it would further complicate the DSM without providing additional insight. Therefore, the third guiding principle is that all two-way negotiation-type interactions between a single pair of parameters are removed in the DSM by recording only net flow of information. Essentially, the DSM represents only the flow of the “purchased item” from seller to buyer and abstracts the negotiation – and the money – from the deal.

4.2.3. Structure of the DSM for ICE Process Analysis

The DSM for the ICE environment consists of 172 design parameters and 682 dependencies among them. The parameters are spread across the 16 disciplines involved in the MDL process (not including the Team Lead and Systems Engineer, whose job is essentially to manage the process defined by the DSM). The entire DSM is shown in Figure 4-2. Because of the size of this DSM, the names of the individual parameters, listed to the left of each row, cannot be fully displayed on the page while still showing the entire matrix.[6] This, however, is not as problematic as it might seem because the purpose of the DSM is to understand the systems-level implications of the parameter dependencies. Instead of focusing on individual parameters, the DSM is organized by subsystem/discipline, and the name of each of the 16 subsystems and disciplines is listed to the right of the DSM. The outlined blocks along the diagonal represent the work internal to each of them, and the off-diagonal elements represent the interdisciplinary information flow that occurs in the design process.

Among the 12 design sessions on which this research is based, seven are considered to be typical according to the criteria established in Chapter 3. This percentage would ordinarily be significantly higher (indeed, by definition). During the data collection period, however, the MDL performed a larger than normal number of advanced or atypical design sessions. Thus, three of the observed MDL sessions were planetary missions involving difficult trajectories or extreme mission environments, and two others were based on new or unfamiliar concepts that involved advanced approaches and/or technologies. For the atypical sessions observed, most of the DSM is still applicable, but certain dependencies and possibly some new parameters would need to be added to fully represent the flow of information. Still, these adjustments are relatively minor once the DSM is constructed according to the guiding principles outlined above. For this reason, the analysis presented throughout the remainder of this chapter is based entirely on the DSM as shown in Figure 4-2 and focuses on the standard DSM for a typical MDL session.

---

[6] An Excel-based version of the full DSM is available from the author upon request by e-mailing avnet@alum.mit.edu. The reader can zoom in to see individual parameters and dependencies or out to see the entire system-level view of the process. The DSM analysis can be reproduced using an Excel add-in called DSM@MIT (Cho et al 2004).

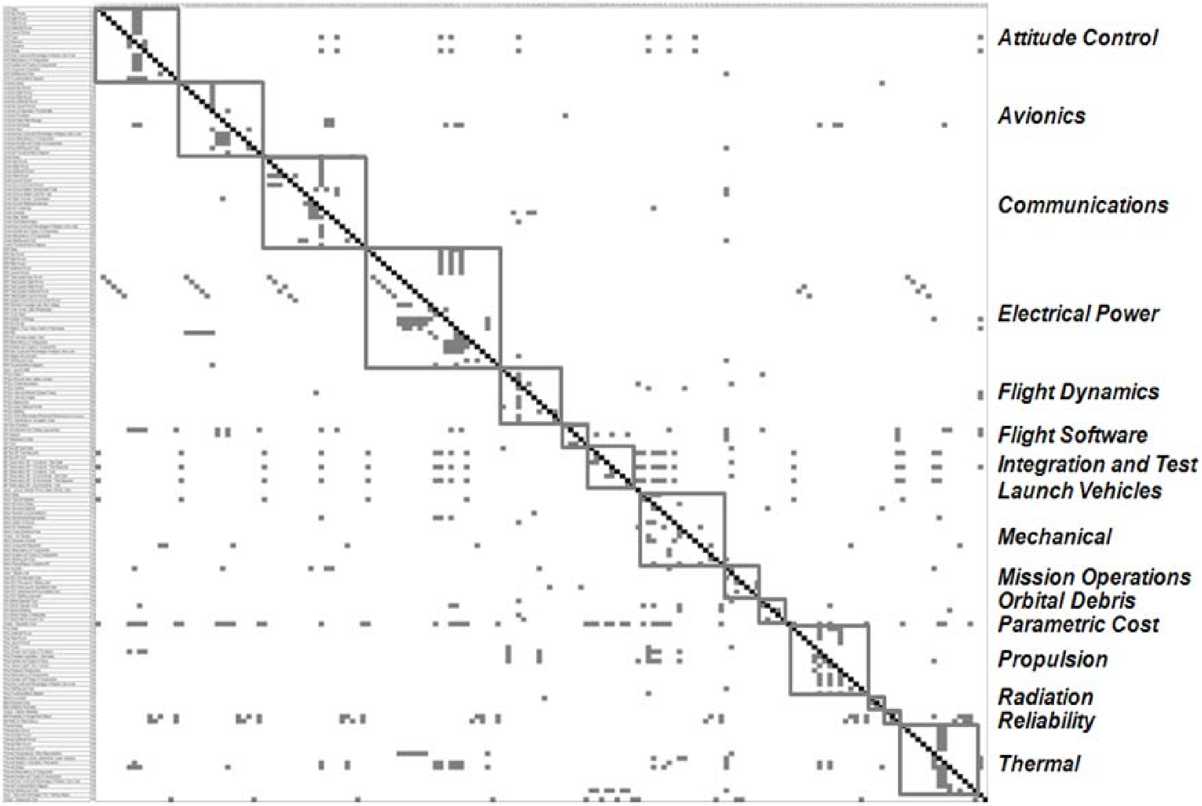


Figure 4-2. **Parameter-Based DSM for the ICE Design Process.** The DSM is organized as an alphabetical sequence of the 16 disciplines involved. The blocks along the diagonal encapsulate the work that is internal to each discipline, and the off-diagonal marks represent information flow across disciplines. Although the names of the individual parameters cannot be fully displayed here, an Excel-based version of the DSM is available upon request by e-mail to avnet@alum.mit.edu.

4.3. The ICE Design Life Cycle

One of the advantages of using the DSM to represent the design process is the set of analytical techniques that can be applied to it. The primary procedure for design process analysis using the DSM is called partitioning. The purpose of partitioning is to reveal the optimal order in which tasks can be completed. In this section, the partitioning procedure is explained, and the features that it reveals about the ICE design life cycle are discussed.

4.3.1. Partitioning the DSM

Partitioning refers to the reordering of the rows and columns in the DSM with the goal of minimizing the number of marks above the diagonal, i.e., to make the matrix lower triangular. The result is an optimal ordering of tasks that reduces feedback and rework to the greatest extent

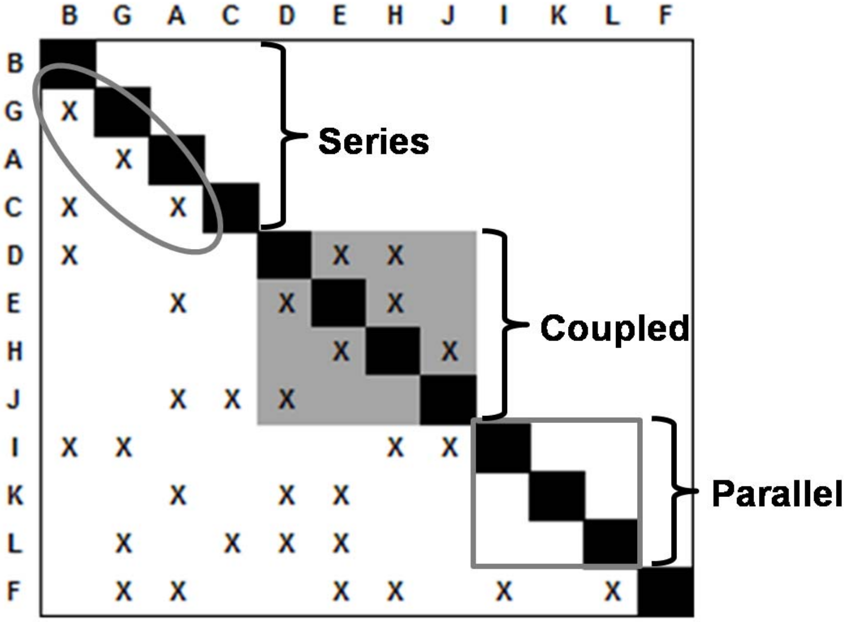


Figure 4-3. Partitioning the Design Structure Matrix. The sequencing of tasks yields an optimal ordering that reduces feedback and rework to the greatest extent possible by minimizing the number of marks above the diagonal. Sets of series, parallel, and coupled tasks can be readily identified in a partitioned DSM.

possible and "maximize[s] the availability of information required at each stage of the design process" (Gebala and Eppinger 1991, p. 229). In a partitioned DSM, three types of tasks can be identified: series, parallel, and coupled (Eppinger 1991). Series tasks are those that must be completed in a specific order. In a pair of series tasks in which *i* depends on *j*, a mark is placed in cell *i*,*j* to indicate that *j* must be done before *i* can be completed. Parallel tasks are those that do not require any information from each other and thus can be completed at the same time. In a pair of parallel tasks within a DSM, no marks exist in either cell *i*,*j* or *j*,*i*. Coupled tasks are those that are inherently linked and must be completed concurrently. In a pair of coupled tasks, a mark exists in both cells *i*,*j* and *j*,*i*, or *i* and *j* are coupled in a more complex way involving other parameters. For example, this occurs if a task *k* requires information from *j*, *j* from *i*, and *i* from *k*. Figure 4-3 shows an example of a partitioned DSM in which sets of series, parallel, and coupled tasks have been identified.

4.3.2. The Partitioned DSM for the ICE Design Process

In this research, the DSM is partitioned using an Excel add-in called DSM@MIT (Cho et al 2004). The partitioned DSM for the ICE environment that results from the application of that

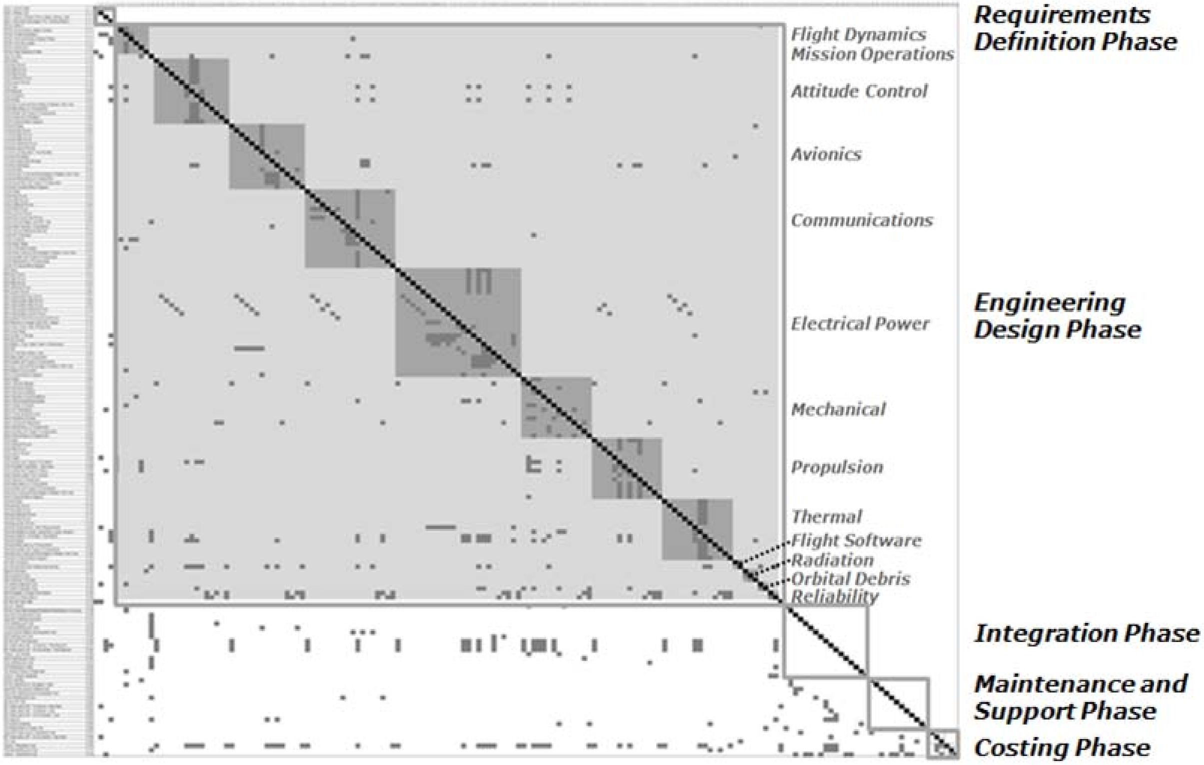


Figure 4-4. Partitioned DSM for the ICE Design Process. The outlines show the phases of the design life cycle, the lightly shaded region highlights the single large coupled block corresponding to the Engineering Design Phase, and the darkly shaded blocks within the larger block contain the engineering tasks of each individual discipline.

tool is given in Figure 4-4. The most striking feature of the DSM is a single large block of coupled tasks, which is lightly shaded in the figure. Interestingly, the tasks that appear in this block are the engineering design parameters, i.e., those that contribute to technical design rather than to programmatic issues like support or costing. Since all of the tasks in the block are tightly coupled, the order among them is immaterial. Thus, the tasks can be organized by discipline to visualize the broader interdependencies among the members of the team. Within the large coupled block, the groups of parameters corresponding to the disciplines are darkly shaded, and the names of those disciplines are indicated to the right of the large block. The particular order in which the disciplines are arranged in the figure places those with the most output parameters (Flight Dynamics and Mission Operations) at the beginning, the spacecraft subsystems next, and the environmental/contextual issues of Radiation, Orbital Debris, and Reliability last. The marks inside the dark blocks represent the engineering design work internal to each discipline, while the marks outside of those blocks show the interdependencies among the disciplines.

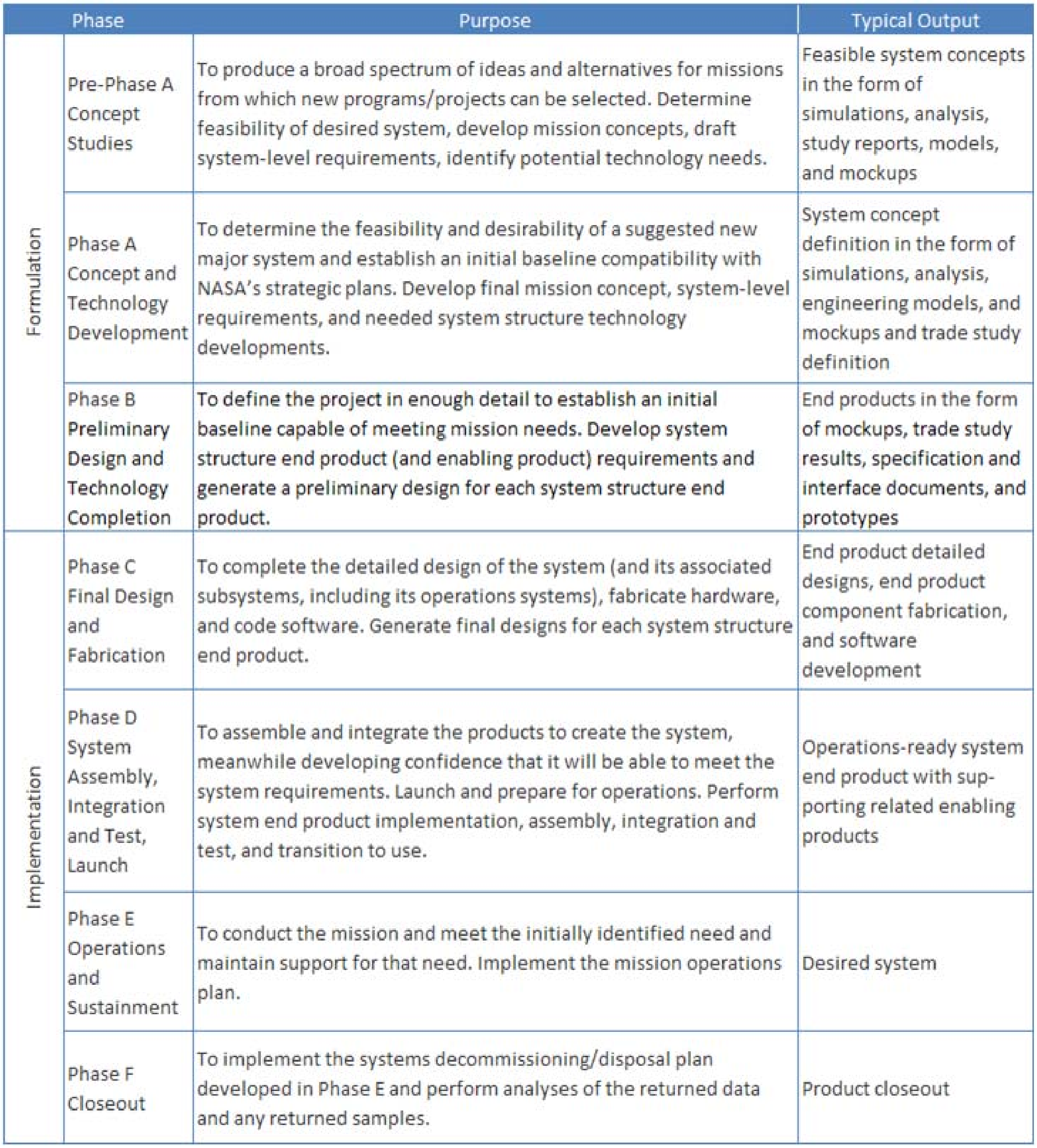
Aside from the large coupled block, the work separates into five distinct phases that map roughly (though not exactly) to Phases A through E of the standard NASA project life cycle (NASA 2007), which is described in Table 4-2. This observation suggests that the ICE environment is structured similarly to full space mission development programs (in fact, the work conducted in an ICE laboratory is actually a pre-Phase A study in itself). The phases of the ICE design life cycle are marked in Figure 4-4 by the outlined boxes along the diagonal and are labeled to the right of the DSM. The phases, as determined by the DSM, are Requirements Definition (~Phase A), Engineering Design (~Phase B and C), Integration (~Phase D), Maintenance and Support (~Phase E), and Costing.[7] The position of the Costing Phase is a notable difference between the structure of the ICE environment and information flow in a full development program. In most programs, cost requirements influence the process from the beginning and present an important design constraint throughout the entire program. Since ICE designs are done at a conceptual level, the upfront constraint is normally just that the mission be of a certain class. At such an early stage of development, this constraint is at least qualitatively similar to specific cost caps in full development programs.

The first phase in the life cycle, Requirements Definition, contains the four major inputs to the process: launch date, required mission life, launch vehicle (which is generally pre-decided by the customer prior to the start of a typical MDL design session), and scientific instruments. The second phase, as described above, contains the engineering design parameters and all of the feedback loops among them. The last few phases primarily include figures of merit for the various subsystems, Integration, Maintenance and Support issues, and Costing (both parametric and grassroots). Since these parts of the work can be implemented sequentially, the issues that remain to be resolved occur in the large coupled block that makes up the Engineering Design Phase. The next section offers a deeper analysis of the coupled block that focuses on the design trades defined by the feedback loops in the process.

---

[7] The Costing Phase is an aggregate of three smaller sets of parallel tasks identified by the partitioning algorithm, so it is actually a sequential phase made up of three sub-phases. These sub-phases are grouped together as one here based on the related content of their parameters.

Table 4-2. NASA Project Life Cylce Phases. Adapted from NASA (2007).



Phase		Purpose	Typical Output
Formulation	Pre-Phase A Concept Studies	To produce a broad spectrum of ideas and alternatives for missions from which new programs/projects can be selected. Determine feasibility of desired system, develop mission concepts, draft system-level requirements, identify potential technology needs.	Feasible system concepts in the form of simulations, analysis, study reports, models, and mockups
	Phase A Concept and Technology Development	To determine the feasibility and desirability of a suggested new major system and establish an initial baseline compatibility with NASA's strategic plans. Develop final mission concept, system-level requirements, and needed system structure technology developments.	System concept definition in the form of simulations, analysis, engineering models, and mockups and trade study definition
	Phase B Preliminary Design and Technology Completion	To define the project in enough detail to establish an initial baseline capable of meeting mission needs. Develop system structure end product (and enabling product) requirements and generate a preliminary design for each system structure end product.	End products in the form of mockups, trade study results, specification and interface documents, and prototypes
Implementation	Phase C Final Design and Fabrication	To complete the detailed design of the system (and its associated subsystems, including its operations systems), fabricate hardware, and code software. Generate final designs for each system structure end product.	End product detailed designs, end product component fabrication, and software development
	Phase D System Assembly, Integration and Test, Launch	To assemble and integrate the products to create the system, meanwhile developing confidence that it will be able to meet the system requirements. Launch and prepare for operations. Perform system end product implementation, assembly, integration and test, and transition to use.	Operations-ready system end product with supporting related enabling products
	Phase E Operations and Sustainment	To conduct the mission and meet the initially identified need and maintain support for that need. Implement the mission operations plan.	Desired system
	Phase F Closeout	To implement the systems decommissioning/disposal plan developed in Phase E and perform analyses of the returned data and any returned samples.	Product closeout

4.4. Design Trades and Interdependent Disciplines

In the design of any complex system, a number of important design trades inevitably must be made throughout the process. In the DSM, these trades are defined by back-and-forth interactions among disciplines that are captured in the off-diagonal marks in the coupled block. In this section, the important design trades are identified through an analysis of the loops in that block, and a team-based DSM is then constructed based on the interdependence among disciplines resulting from the design trades.

4.4.1. Loop Analysis

During a typical MDL design session, most of the week's work occurs in the large coupled block that corresponds to the Engineering Design Phase of the life cycle. Before and after this phase, the work is essentially a straightforward sequence of tasks performed either in series or in parallel. Not only does that phase contain the bulk of the work for 13 of the 16 subsystems and disciplines,[8] but it also represents the work for which interactions among disciplines is most important.

The purpose of this subsection is to elucidate the most important interactions among the subsystems and disciplines. These interactions can be found through a deeper analysis of the coupled block. During the sequential phases, information flow involves the simple delivery of parameter values from one discipline to another. The loops in the Engineering Design Phase, however, represent the critical design trades by which certain disciplines in the team are tightly coupled. Thus, the goal of this analysis is not to catalog every loop in the DSM (of which there are at least several million) but rather to determine the interdependence among disciplines resulting from the design trades that the loops represent. See Appendix C for a complete list of the 187 loops with a length of five parameters or less.

The loop analysis procedure described here represents a new way of looking at the problem of interdependencies in a system. Although finding loops is a step in DSM partitioning (Gebala and Eppinger 1991), the algorithm is concerned only with determining whether parameters are coupled together in loops and not with the specific content of those loops. Because of the rapid rate at which the number of loops increases with the size of the network, a full loop analysis is not generally applied to DSMs as large as the one constructed for the ICE design process. To account for this complexity, the loop analysis procedure used here focuses on the shortest loops in the network and is dependent on the content of the parameters in each loop. Thus, this procedure is not meant as a complete analysis of all loops in the DSM but rather as a means of identifying the interdependencies among disciplines in the design process. For this reason, the

---

[8] Launch Vehicles, Integration and Test, and Parametric Cost are not found in any loops because the first is an input to the process and the latter two outputs.

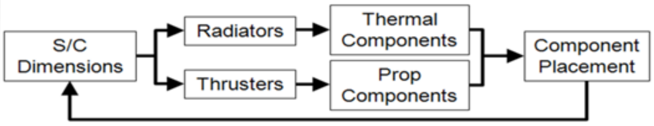
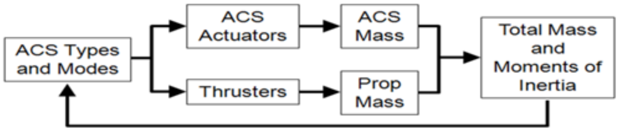
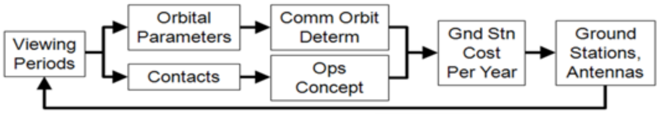
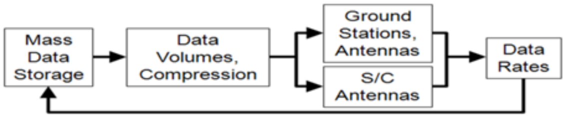
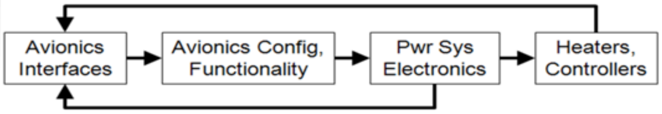
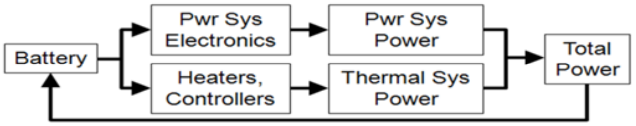
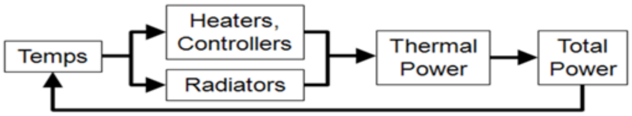
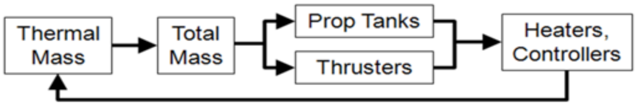
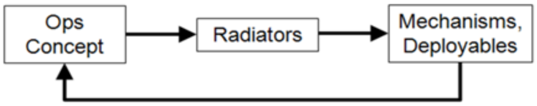
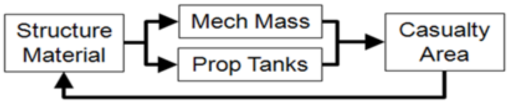
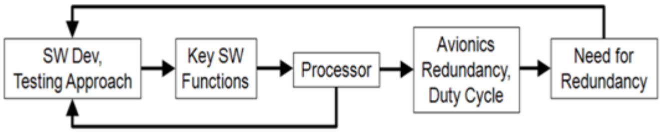
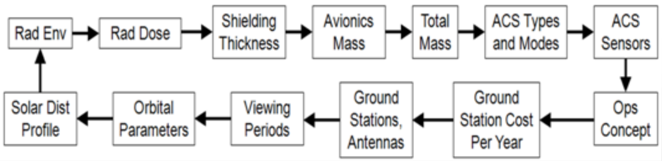
procedure is generalizable only for cases in which the purpose is similar to the one here – to identify interdependencies among certain sets of parameters (in this case, disciplines in the design team).

To determine the tightest interdependencies in the process, the loop analysis begins with the shortest loops – those that contain only three or four parameters. Once these three- and four-parameter loops are identified, the ones that represent the same type of information exchange are collapsed into a single type or class of loop. For example, Electrical Power collects power requirements from all other subsystems for several power modes. If a set of loops involving Electrical Power are identical except that they include trades for different power modes, they are classified as a single "loop type." Based on this procedure, 10 of the 13 disciplines in the large coupled block are represented among the three- and four-parameter loops in the DSM. To find the critical trades for the three remaining disciplines in the coupled block, it is then necessary to look at some loops that are longer than four parameters.

The disciplines in the coupled block that are not found in any three- or four-parameter loops are Flight Dynamics, Reliability, and Radiation. The first two of these are captured in five-item loops. Since the purpose of this procedure is to determine the tightest coupling among disciplines, these five-item loops complete the analysis for Flight Dynamics and Reliability. At this point, it is only necessary to find longer loops containing parameters from Radiation. As it turns out, the shortest loop containing trades on Radiation is 13 parameters long and involves seven disciplines. This result agrees with observation. In a typical MDL session, the Radiation engineer is actively involved at the start and at the conclusion of the work. At those times, the relevant design trades are resolved, but regular support from the Radiation engineer is not critical throughout most of the session.

This loop analysis results in 13 general loop types representing the classes of feedback that occur during the resolution of critical design trades. These loop types are shown in Table 4-3. The name assigned to each loop type in the first column is intended to describe the primary design trade that is resolved among the parameters in that loop. In the second column, the structure of each of the 13 loop types and the names of the parameters involved in each are shown. In the third column of the table, the disciplines that are tightly coupled as a result of the design trades are identified. Arrows (→) indicate the direction of information flow among the

Table 4-3. Feedback Loops in the ICE Design Process. The 13 types of loops represent the classes of feedback that occur as a result of design trades in a typical ICE design session. The coupling of disciplines implied by each loop type is indicated in the rightmost column.

Loop Type	Loop Structure	Coupled Disciplines
Spacecraft Bus Loop		**Mechanical ↔ Thermal;** **Mechanical ↔ Propulsion**
Propulsion Sizing Loop		**Propulsion ↔ Mechanical**
Stabilization Loop		**Attitude Control ↔ Mechanical;** **Attitude Control → Propulsion →** **Mechanical → Attitude Control**
Ground Segment Loop		**Flight Dynamics ↔** **Communications;** **Communications ↔ Mission** **Operations**
Data Loop		**Avionics ↔ Communications**
Power System Electronics Loop		**Avionics ↔ Electrical Power →** **Thermal → Avionics**
Power Loop		**Electrical Power ↔ Thermal**
Electrical Heating Loop		**Thermal ↔ Electrical Power**
Propulsion Thermal Control Loop		**Thermal → Mechanical →** **Propulsion → Thermal**
Radiator Operation Loop		**Mission Operations → Thermal →** **Mechanical → Mission Operations**
Reentry Loop		**Mechanical ↔ Orbital Debris;** **Mechanical → Propulsion →** **Orbital Debris → Mechanical**
Computing Reliability Loop		**Flight Software ↔ Avionics →** **Reliability → Flight Software**
Radiation Shielding Loop		**Radiation → Avionics →** **Mechanical → Attitude Control →** **Mission Operations →** **Communications → Flight** **Dynamics → Radiation**

disciplines, and double arrows (↔) denote complex two-way negotiations involving several aspects of the work of two disciplines.[9] To understand how the coupled disciplines are determined, consider the structure of the Stabilization Loop as an example. According to this loop type, Attitude Control and Mechanical trade back and forth with each other in the process of resolving several parameters. In addition, Attitude Control (types and modes) influences Propulsion (thrusters), which in turn affects Mechanical (total system mass and moments of inertia). Mechanical then affects the design of the Attitude Control subsystem. Thus, the coupling of disciplines for that loop type is: Attitude Control ↔ Mechanical and Attitude Control → Propulsion → Mechanical → Attitude Control.

4.4.2. Critical Design Trades

The loop types shown in Table 4-3 do not merely depict feedback among technical parameters in the design, but they also represent the most important interfaces over which the members of the team interact. Each of these loop types specifies a certain class of design trade that requires purposeful interaction among two or more members of the design team. The following paragraphs briefly describe each of these trades because they are the source of the tightest coupling among disciplines in the design process.

The Spacecraft Bus Loop involves the placement of certain components within the size limits of the bus. The Propulsion Sizing Loop reveals the important effect that Propulsion has on system mass. The Stabilization Loop results from the fact that Attitude Control places certain requirements on Propulsion, which affects mass and moments of inertia. These, in turn, affect Attitude Control. The Ground Segment and Data Loops deal with data transmission and storage. In the former, the tradeoff between the spacecraft architecture, its orbit, and the ground segment is considered. In the latter, the trades associated with data storage are captured.

The Power System Electronics Loop describes a particularly important trade that is made early in each design session – whether the Power System Electronics (PSE) box is managed by Avionics or Electrical Power. This is an important issue because it dictates the types of interactions that are then needed between these two disciplines throughout the remainder of the session. The Power Loop captures another important issue in spacecraft design. Not only does

---

[9] These two-way trades are different from direct back-and-forth negotiations involving just two parameters, which (as discussed previously) have been abstracted from the DSM.

Electrical Power track power requirements and allotments from other subsystems, but this discipline also includes a subsystem design with its own power requirements. Because the Electrical Power subsystem has its own internal power requirements, a tradeoff exists among power generation, energy storage, and the subsystem's own power requirements. The Electrical Heating Loop represents the trade that is made between power generation capacity and the heat that is generated as a by-product. Similarly, the Propulsion Thermal Control Loop demonstrates the tradeoff between providing thermal protection for the tanks and thrusters and maintaining the overall mass of the system.

The Radiator Operation Loop describes a less intuitive but still interesting trade of only three parameters that spans several aspects of the design. The Mission Operations concept (or ops concept) imposes certain requirements on the design of the radiators by Thermal, and the radiators affect the design and use of the mechanisms on the spacecraft. The operation of the mechanisms, of course, influences the ops concept. The Reentry Loop demonstrates an important trade regarding spacecraft end-of-life. The casualty area is the surface area of spacecraft elements that could survive reentry into Earth's atmosphere intact and thus pose a hazard on the ground. This loop demonstrates the tradeoff between limiting this hazard and designing the spacecraft to best achieve mission objectives. The Computing Reliability Loop contains two closely related design trades. First, a three-item loop shows the trade that is made between software requirements and Avionics' processor design. Second, a five-item loop demonstrates that Reliability does not simply impose upfront requirements or determine overall mission reliability but also places constraints on the design of Flight Software.

Finally, the Radiation Shielding Loop is important because it demonstrates that the role of the Radiation discipline is qualitatively different from the other disciplines involved in the design process. Early in the session, Radiation receives a solar distance profile from the Flight Dynamics engineer based on the intended trajectory and orbit of the spacecraft. After the Radiation engineer determines the level of expected exposure for various amounts of shielding, this information is incorporated into the sizing of the Avionics subsystem, which is particularly sensitive to radiation issues. Then, the Radiation engineer generally leaves the room and does not participate actively in the session again until the end. Over the next few days of design work, one factor influences the next in a series of interactions that involves Mechanical, Attitude

Control, Mission Operations, and Communications. By the end of the week, this cascade reaches Flight Dynamics, and the Radiation analysis is then affected accordingly.

A more detailed discussion of space systems design is beyond the scope of this thesis but is provided by Wertz and Larson (1999). Whereas that extensive volume includes in-depth coverage of all aspects of space mission design from experts in each discipline, the analysis presented here offers an accessible systems-level overview of a particular space mission design context in terms of the most important cross-disciplinary interactions in the process. In the next section, these interactions across disciplines are formally integrated in a team-based DSM representation, and the resulting sets of interdependent disciplines are identified.

4.4.3. A DSM Representation of Interdependent Disciplines

The loop types described in the previous subsection represent the critical trades of technical issues in the design process. Because of the significance of these interactions, the loop types also represent the most important interactions among members of the design team. Recall that the rightmost column of Table 4-3 shows the disciplines that are coupled together as a result of each of the loop types. These important dependencies can be represented by placing the appropriate marks in a 16 *x* 16 matrix with disciplines in the rows and columns. Continuing this process for all 13 loop types results in a team-based DSM of 16 subsystems/disciplines and 32 interdependencies among them. Information flow that is not involved in any of the loops occurs relatively easily from one discipline to another and thus does not constitute tight interdependence between the disciplines involved. Therefore, this type of information flow is not included in the team-based DSM.

Because the team-based DSM represents people rather than tasks, partitioning is not the appropriate means of analysis. Instead, the team-based DSM is analyzed with the goal of identifying logical groupings of interdependent entities. The method of identifying groupings in a team-based DSM is known as clustering. In practice, this term actually refers to a broad category of algorithms that use different but related techniques to divide a DSM into groups of entities that are tightly connected internally and more sparsely connected externally. In this research, the particular clustering method used is called the Newman-Girvan community structure algorithm (Newman and Girvan 2004). In the terminology of this algorithm, a cluster is called a community, and the organization of the DSM into those communities is called the community structure.

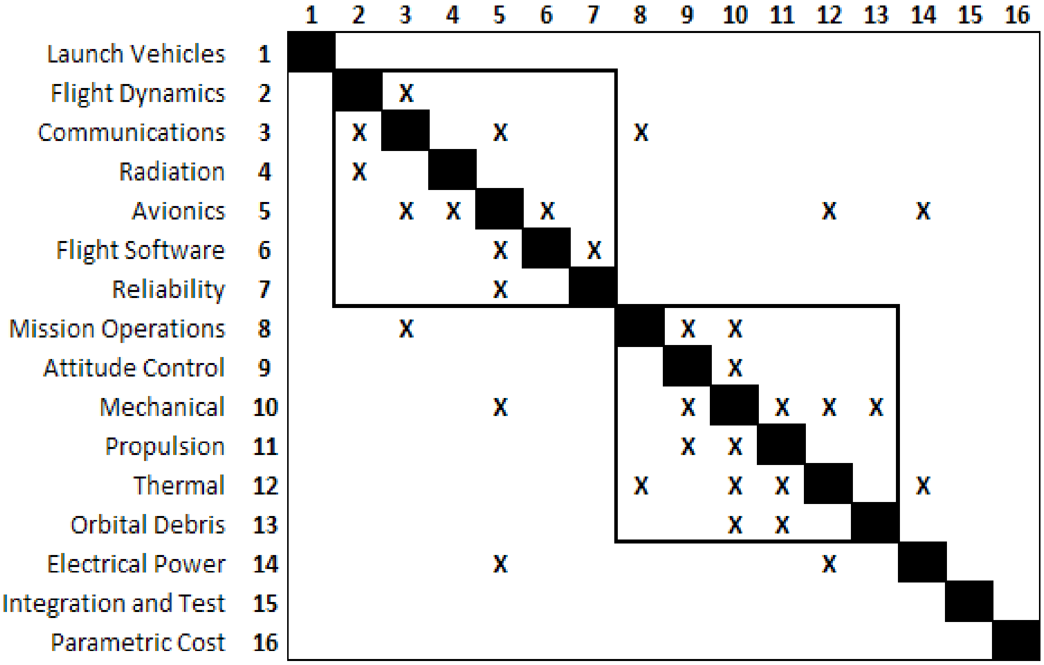
The Newman-Girvan algorithm is based on the principles of graph theory and thus treats the DSM as an adjacency matrix of a network.[10] The method is particularly useful because it is able to determine the optimal number of communities (clusters) in which to split the DSM by calculating a metric of modularity, $Q \in [0,1]$. Modularity is the proportion of dependencies in the network that are internal to the communities adjusted according to the same ratio computed without consideration to community structure. The number of communities that maximizes Q corresponds to the optimal community structure. For most real-world systems, Q generally is between 0.3 and 0.7 (Newman and Girvan 2004). See Appendix B for a description of the Newman-Girvan algorithm and the mathematical definition of modularity.

In this research, the team-based DSM is clustered using the Newman-Girvan algorithm as implemented in the network analysis software package UCINET (Borgatti et al. 2002). The resulting clustered team-based is given in Figure 4-5. It is important to note that the results do not imply that each cluster can operate independently of the others. Although the clustering of the team-based DSM maximizes Q, that value is still only $Q = 0.306$. This degree of modularity is high enough to constitute a meaningful division but is sufficiently low to indicate that cross-cluster coordination remains important. Moreover the dependencies in the team-based DSM comprise a relatively small subset of all parameter dependencies in the process. Much of the information flow is not necessarily bound in the design trades represented by the 13 loop types. For example, Launch Vehicles, Integration and Test, and Parametric Cost are included in this DSM even though there are not any dependencies in the rows and columns corresponding to those disciplines. This is appropriate because it highlights the reality that these disciplines do not exchange information with other disciplines within the context of the critical design trades represented by the 13 identified loop types but that they still make up an important part of the process.

Until this point, the analysis has focused on how the design work is done given the high degree of interdependence in the large coupled block of the Engineering Design Phase. In the

---

[10] An adjacency matrix is a matrix equivalent of a network in which the value of cell *i,j* denotes the existence of an edge (connection) between nodes (entities) *i* and *j*.



		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Launch Vehicles	1	■															
Flight Dynamics	2		■	X													
Communications	3		X	■		X			X								
Radiation	4		X		■												
Avionics	5			X	X	■	X						X		X		
Flight Software	6					X	■	X									
Reliability	7					X		■									
Mission Operations	8			X					■	X	X						
Attitude Control	9									■	X						
Mechanical	10					X				X	■	X	X	X			
Propulsion	11									X	X	■					
Thermal	12								X		X	X	■		X		
Orbital Debris	13										X	X		■			
Electrical Power	14					X							X		■		
Integration and Test	15															■	
Parametric Cost	16																■

Figure 4-5. Clustered Team-Based DSM for the ICE Design Process. Each mark indicates that the discipline in the row requires information from the discipline in the column due to one or more of the critical design trades.

next section, a method of controlling for that complexity by making certain starting assumptions at the outset of the design process is introduced.

4.5. Defining an Iterative Process for the ICE Environment

Given the highly cyclic nature of the design process as revealed by the coupled block in the DSM, it is unclear where exactly the design work should begin. In the partitioned DSM shown in Figure 4-4, the parameters in the coupled block are arranged according to discipline, and the order of the disciplines within that block is based on the researcher's qualitative understanding of how the work is normally conducted in the ICE environment. This, however, cannot be taken to be the order in which the tasks within the coupled block should be executed. In fact, in the partitioned DSM, there is no such required ordering of tasks because the coupled parameters need to be resolved together. Still, for the work to proceed from the Requirements Definition Phase to the Engineering Design Phase, it is necessary to determine a point at which the process should "enter" the loop. A DSM analysis tool called tearing can be used to determine the optimal place to begin the design work. In this section, the tearing procedure is explained, the method of identifying starting assumptions is described, and the implications for the design process are discussed.

4.5.1. Tearing the DSM

The goal of tearing a DSM is to determine the dependencies that, if removed, would result in a lower triangular matrix depicting a sequential design process consisting entirely of series and parallel tasks. The precise method used for tearing depends on the goals that one is trying to achieve, i.e. which types of dependencies are best removed (Gebala and Eppinger 1991), but the goal is always to determine a starting point for the design process. After the chosen dependencies are removed (or torn), the DSM is repartitioned "to find an initial ordering to start the iteration" (Gebala and Eppinger 1991, p. 229). Thus, the torn marks represent the starting assumptions that can be made to optimize the process. Of course, once the series of tasks defined by the torn DSM is completed, the entire process must be iterated to refine the starting assumptions.

Figure 4-6 demonstrates the procedure for tearing the DSM. Starting with the partitioned example DSM in Figure 4-3, three marks are identified as candidates to be torn. Although one mark remains above the diagonal in this example, it ends up below the diagonal after the torn DSM is repartitioned. In the resulting DSM, the marks that were torn now appear above the diagonal and are colored in red font to indicate that they are the starting assumptions that can be made before the start of the first iteration. The number of subsequent iterations must then be determined based on the level of fidelity required in the final design. Regardless of the exact number of iterations chosen, however, this procedure facilitates a sequential process for implementing the project so that the starting assumptions can be refined and improved in a well-defined and systematic way.

4.5.2. Design Budgets and Other Starting Assumptions to the ICE Process

Unlike the simple example shown in Figure 4-6, the full partitioned DSM for the ICE environment does not have a small and well-defined number of marks above the diagonal that can be quickly found, torn, and identified as starting assumptions. Given the large number of marks above the diagonal, it is difficult to determine which ones should be removed to yield a sequential process. Removing all dependencies above the diagonal would mean making an

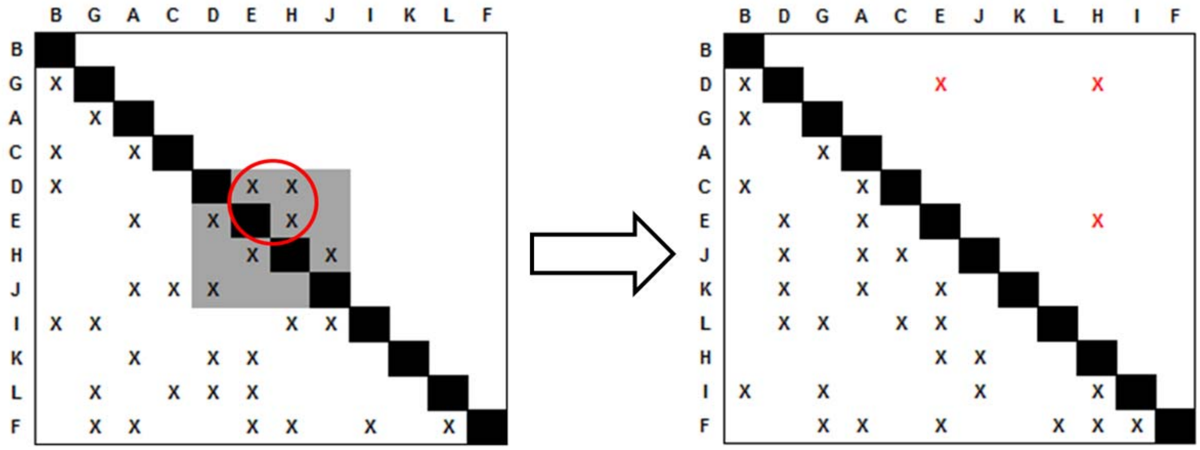


Figure 4-6. Tearing the DSM. In the partitioned DSM, certain marks can be removed that will cause the DSM, once repartitioned, to become lower triangular, which corresponds to a sequential design process. The torn marks, colored in red in the torn DSM, correspond to the starting assumptions to the process.

unreasonable and unnecessary number of assumptions. Furthermore, it is critical to ensure that the right parameters are chosen because the values for some parameters can be assumed more easily and with greater confidence than for others.

Upon inspection of the coupled block in the DSM, it is readily obvious that each of the disciplines collects several pieces of information from many other disciplines. This can be seen from the horizontal rows of marks across the coupled block. A closer look at the content of those dependencies shows that information gathered by each discipline engineer is of the same type across all disciplines from which this information is collected. For example, in the rows for Electrical Power, that discipline engineer collects power requirements for each of the other subsystems. The totals that result from collecting this information usually must stay within a specified range for the particular system. These totals are called design budgets. The values for these budgets can often be assumed at the start of the first iteration using historical data from similar past missions (indeed, this is standard practice in the Mission Design Laboratory).

In general, spacecraft design involves certain basic types of design budgets: the power budget, the mass budget, the propellant budget, the reliability budget (Reeves 1999), the link budget (Dietrich and Davies 1999), and the pointing and mapping budgets (Wertz 1999). Figure 4-7(a) shows the partitioned DSM with three of the budgets identified. In addition to these main design budgets, several other collections of parameters behave similarly to design budgets in the structure of the DSM. Although these sets of parameters are not traditional budgets in that they

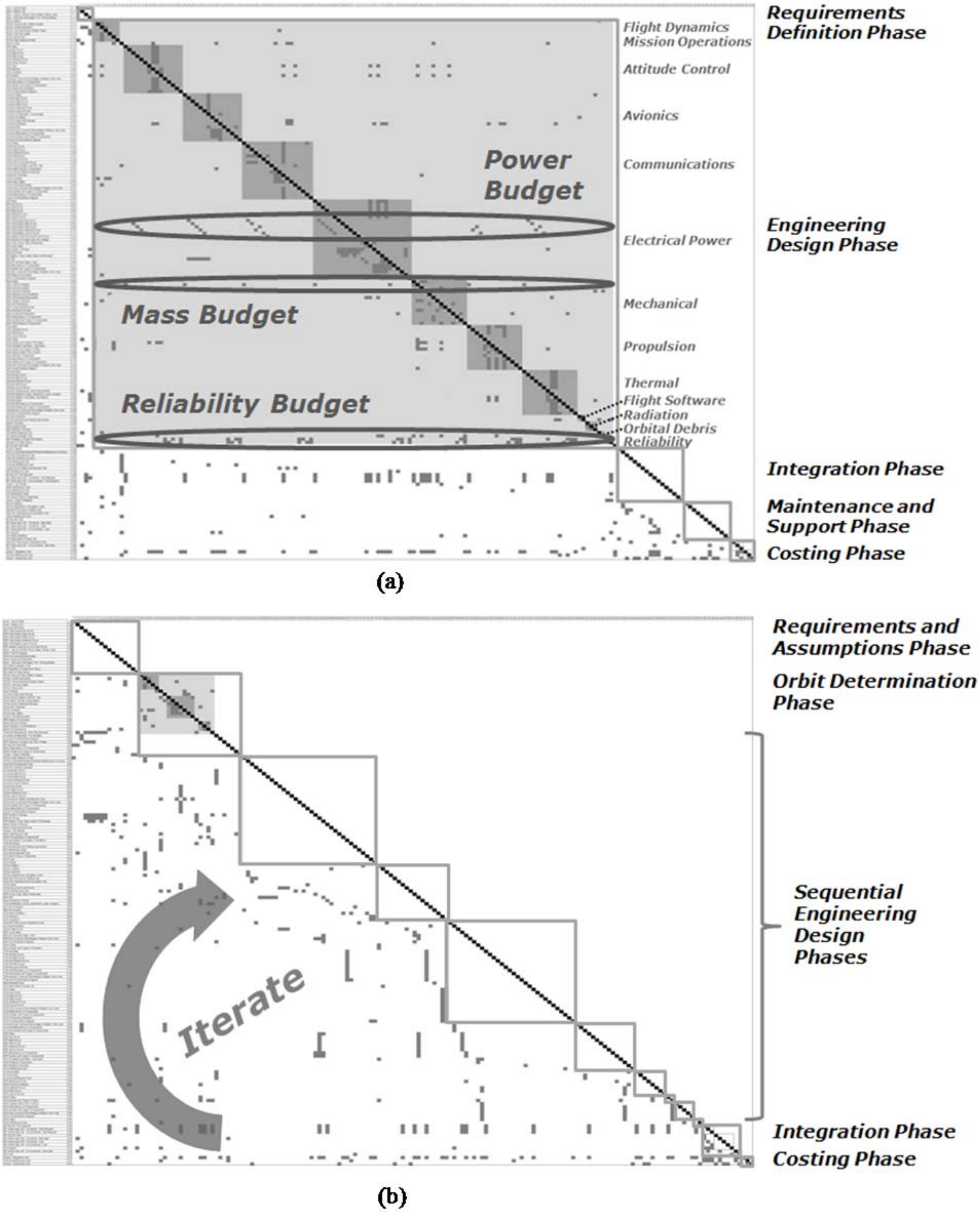


Figure 4-7. Tearing the DSM for the ICE Design Process. (a) Partitioned DSM with Design Budgets Identified. (b) Torn and Repartitioned DSM.

do not require the summing of numbers to stay below a certain total, they are uniform pieces of information about which starting assumptions can be made. These other “budgets” are

- Avionics Interfaces – components for which electronic interfaces must be defined,
- Mission Operations Hardware – physical spacecraft components that affect Mission Operations,
- Software Development Factors – components that affect required software,
- Thermal Design Factors – components that affect the thermal design, and
- Casualty Area Factors – attributes and components that could pose a hazard on the ground after reentry.

Figure 4-7(b) shows the resulting torn DSM after removing the design budgets while keeping in place other dependencies about which assumptions cannot be made as easily. As the figure shows, the process becomes almost entirely sequential among the remaining 495 dependencies. In the torn and repartitioned DSM, the Requirements Phase has been renamed the Requirements and Assumptions Phase to emphasize the need to define those assumptions at the start of the work. The Engineering Design Phase has been split into a series of sequential phases, and the Maintenance and Support Phase has been subsumed into those phases. Still, although the torn DSM shows all of the design phases, it represents only the start of a highly iterative process. The circular arrow in Figure 4-7(b), labeled “Iterate,” demonstrates that the sequential process defined here is merely a single iteration of the design. As discussed previously, subsequent iterations must then be made to refine the assumptions and to ensure that they are correct for the specific system being designed.

Tearing the DSM reveals that most of the interdependent information in the design process is in the form of established and reasonably predictable design budgets and other collections of parameters. Even though most aspects of the design are tightly coupled together, this coupling can be managed by beginning the work with a set of starting assumptions for each discipline. This, however, does not complete the tearing analysis because a small coupled block still remains in the torn DSM. This block, named the Orbit Determination Phase, consists of most of the parameters needed to specify the spacecraft’s orbit. The meaning and implications of this block of dependencies are discussed in the next subsection.

4.5.3. The Interdependent Core of Space Mission Design (or The ICE Core)

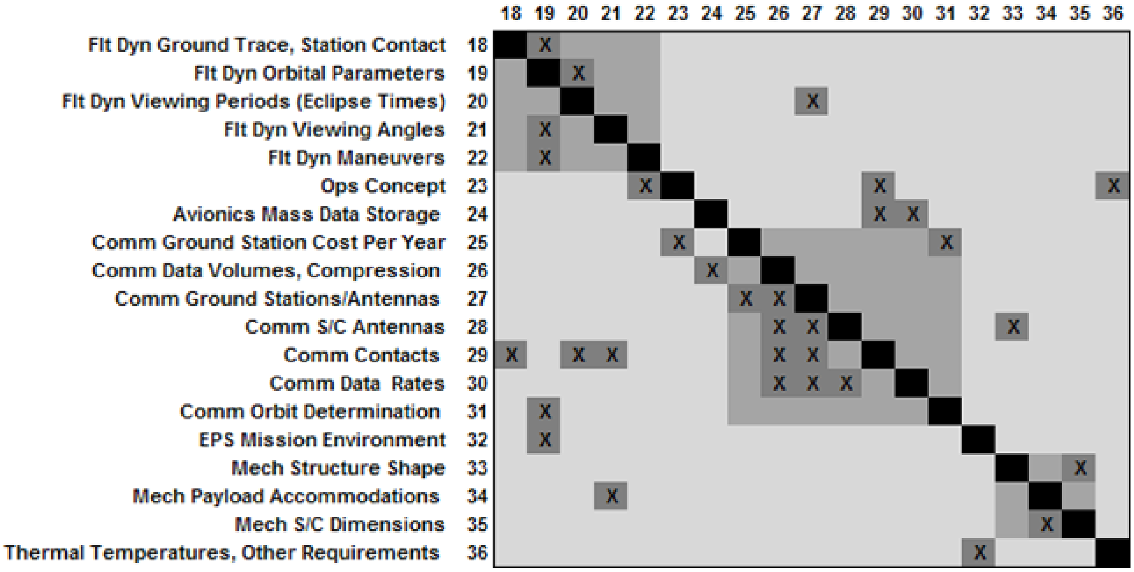
As shown by the existence of the small coupled block in Figure 4-7(b), the ICE design process cannot begin immediately from the sequence revealed by the torn DSM. That small block contains perhaps the most important trades in the entire design process because they cannot be resolved by making starting assumptions and then iterating. Instead, they must be resolved before the sequence of design iterations can begin. For this reason, that block is viewed as an interdependent core of the ICE design process. Figure 4-8 shows a magnified view of the core including its 19 parameters and 32 dependencies. The trades in the core are generally the same ones identified by the Ground Segment Loop and the Data Loop in Table 4-3, and they imply that the central issues that must be resolved early in the process are the spacecraft's location in space and how it communicates with the Earth.

Inspection of the parameters in the core reveals that the relevant subsystems and disciplines are Flight Dynamics, Mission Operations, Avionics, Communications, Electrical Power, Mechanical, and Thermal. Some of these, however, are not as tightly bound in the core as others. First, consider the role of Electrical Power and Thermal in this block. The parameters listed for them are mission environment and temperatures, respectively. Therefore, these two subsystems are in the core only because the orbit influences environmental effects on the spacecraft, but most of those subsystems' engineering design parameters are not in the core. Next, consider Avionics and Mechanical. Mass data storage, provided by Avionics, is an important part of the Data Loop, but it is the only parameter from that subsystem present in the core. The other two important design issues normally handled by Avionics, electronic interfaces and the spacecraft's processor, do not appear in the core. For Mechanical, issues related to spacecraft shape are found in the core, but the parameters that determine system mass and mechanisms are found elsewhere in the full torn DSM.

Among the remaining three disciplines – Flight Dynamics, Mission Operations, and Communications – all engineering design parameters are bound in the core.[11] This implies that there are some especially important trades that need to be made among these disciplines before the first iteration of the full process can begin. From observations of sessions in the Mission

---

[11] For Mission Operations, all aspects of engineering design are captured in a single parameter called ops concept, and all other parameters for that discipline are related to staffing and cost.



		18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Flt Dyn Ground Trace, Station Contact	18	■	X																	
Flt Dyn Orbital Parameters	19		■	X																
Flt Dyn Viewing Periods (Eclipse Times)	20			■							X									
Flt Dyn Viewing Angles	21		X		■															
Flt Dyn Maneuvers	22		X			■														
Ops Concept	23					X	■						X							X
Avionics Mass Data Storage	24							■					X	X						
Comm Ground Station Cost Per Year	25						X		■						X					
Comm Data Volumes, Compression	26							X		■										
Comm Ground Stations/Antennas	27								X	X	■									
Comm S/C Antennas	28									X	X	■					X			
Comm Contacts	29	X		X	X					X	X		■							
Comm Data Rates	30									X	X	X		■						
Comm Orbit Determination	31		X												■					
EPS Mission Environment	32		X													■				
Mech Structure Shape	33																■		X	
Mech Payload Accommodations	34				X													■		
Mech S/C Dimensions	35																	X	■	
Thermal Temperatures, Other Requirements	36															X				■

Figure 4-8. The Interdependent Core of Space Mission Design. Although the ICE core contains seven of the 16 disciplines in the design process, the engineering design work of only three is fully bound in this block. These three disciplines are Communications, Flight Dynamics, and Mission Operations. Thus, the primary trade decided among these three disciplines – namely orbit determination to establish communication links – must be resolved before the first iteration of the torn DSM can begin.

Design Laboratory and discussions with the team, it is apparent that Communications already works with Flight Dynamics at the beginning of each session to determine orbital parameters and their effect on the Communications subsystem design. Thus, the engineers for these two MDL disciplines have either implicitly or explicitly realized the need to resolve these important orbit determination issues. The DSM analysis, however, both formally demonstrates the importance of this interaction and provides some additional insights that can improve it. For instance, the dependencies in the core indicate that Mission Operations should be involved in this discussion from the start. Also, the core shows the effects that this early trade can have on other disciplines, namely Avionics, Mechanical, Electrical Power, and Thermal. Finally, the torn DSM shows that the iterations of the design can proceed sequentially if and only if orbit determination issues are resolved prior to the start of the first iteration.

Thus, the insight that comes from the existence and structure of the interdependent core of the DSM is that the trades bound in this small coupled block should serve as the first step in the design process. In some cases, tailoring the DSM to the specifics of an individual design session could reveal somewhat different implications depending on the important issues for the particular system under consideration. Despite the individual character of each mission, this analysis demonstrates a structure for the design process that can be applied to any MDL session provided that the necessary adjustments are made to the DSM on a case-by-case basis.

Furthermore, some of the results presented here could apply in other contexts beyond the MDL. For example, the Team Lead of the MDL's partner design center, the Instrument Design Laboratory (IDL), has suggested that a similar central issue seems to exist in that lab's design process. In that case, the central issue/discipline is Optical. Thus, a DSM constructed for the IDL may reveal a similar core of interdependent disciplines structured around certain design trades in which Optical is a central discipline. Of course, this can only be determined if such a DSM were to be constructed for that design setting. In still other design settings, different issues may arise as the central ones. In the next section, the applicability of this work to other settings is discussed, and the key aspect of the people in the process is introduced.

4.6. Implications for Applying the DSM to Space Systems Design

This chapter has presented a series of insights about the space systems design process obtained through the application of a systems-level model that codifies and aggregates the tacit knowledge of all engineers involved. This matrix-based model uses interdependencies among subsystems and disciplines to reveal a general structure for the design process. The process defined by the DSM consists of five phases that map roughly to those established by NASA for full space systems development programs. The model shows that the actual design work takes place in the tightly coupled Engineering Design Phase. Based on loops found in this large coupled block, clusters of interdependent disciplines have been identified. These groupings can be used to facilitate the resolution of critical design trades. Finally, a set of starting assumptions provides an initial point from which to begin the process and allows the design to proceed through a series of well-defined iterations. The number of iterations actually executed depends on the level of fidelity required in the design. This process of iterating on a set of sequential design steps can begin only after Communications, Flight Dynamics, and Mission Operations have resolved certain key design trades related to the trajectory and location of the spacecraft.

The product of the work presented in this chapter is not only the first application of the DSM methodology to ICE, but it is also the first full DSM for the space mission design process in general. Therefore, the research provides a basis for structured process analysis in the design and development of space systems of any size and scope. Because the ICE environment is representative of a full development program, the overall structure of the DSM can be expected to be similar for those settings. Although some changes inevitably would have to be made to account for the greater level of detail and the particular area of focus in each program, the DSM presented in this chapter may serve as a template to simplify the daunting task of DSM construction for this type of complex systems engineering endeavor.

Before the DSM can be applied to other settings, though, the implications of the analysis must be considered in the context of actual design sessions in practice and not just as a model of a typical process. The precise manner in which the DSM is implemented will depend on a number of real-world factors affecting each individual design session. In some cases, the customer team might have certain preferences for either the sequence of activities in the session or the specific issues on which the design should focus. Other times, the management style of the particular Team Lead or Systems Engineer for a given session could affect the implementation of the DSM. Finally, the dynamics of the design team and the way the people on the team work best – individually and collaboratively – should be made a top priority in incorporating the insights from this chapter into the design process.

Ultimately, these issues come down to a single inescapable reality – that the work modeled from a technical perspective in this chapter is actually done by human beings. Each person has his or her own distinct knowledge and perspective that he or she brings to the design process. For this reason, the recommendations based on this analysis must take this reality into account. Reaching these recommendations requires an analysis of the design team to complement this chapter's analysis of the design process. Accordingly, the next chapter proposes a model for analyzing shared knowledge in engineering design teams.

[This page intentionally left blank]

Chapter 5

The Structural Approach to Shared Knowledge

In the previous chapter, a method for analyzing the technical design process in the Integrated Concurrent Engineering (ICE) design environment was presented. An important part of that analysis was the identification of team roles that are tightly interdependent based on technical information flow in the design process. In this chapter, the discussion shifts to the commonalities in how members of the team *think* about the work. The first section provides a brief overview of various approaches to shared knowledge, including the one proposed here. In the second section, a model of shared knowledge in teams is developed in detail. The discussion includes a proposed metric for shared knowledge and a procedure for building a network of shared knowledge based on that metric. Then, the third section introduces a method of measuring the dynamics of shared knowledge, i.e., how shared knowledge changes over time. The fourth section tests the model through a demonstration of the relationship between the dynamics of shared knowledge and several technical attributes of the system: mission concept maturity, system development time, launch mass, and system cost. After that, the fifth section presents a sensitivity analysis showing that the results are not subject to specific choices made in the modeling methodology. Finally, the last section of the chapter considers the meaning and implications of shared knowledge. This section contains more questions than answers and thus frames the discussion on the connection between shared knowledge and design process presented in Chapter 6.

5.1. A New Approach to the Study of Shared Knowledge in Teams

The approach taken in this research to measuring shared knowledge in teams is fundamentally different from the three other approaches used in the literature, but it also draws on each of them in its formulation. These three approaches are called collective, holistic, and naturalistic. The collective approach is based on the mainstream view employed in the literature on shared mental models. Although most of that literature focuses on dyads (teams of two), the collective approach to shared knowledge refers to the calculation of a *team mental model* by taking the average of shared mental models in every dyad. The intuitive appeal of this approach is that it attempts to measure actual cognition of each person and scale it to the entire team. The holistic approach explicitly attempts to account for the fundamental drawback of the collective approach. This disadvantage is essentially that the collective approach treats the whole, in a sense, as merely the sum of its parts. The holistic approach, on the other hand, infers *team cognition* from team behavior in much the same way that one might infer individual cognition from individual behavior. The analysis is often based on counts of spoken words, but it does not consider any pair-wise knowledge sharing and has only been applied to three-person teams in controlled laboratory settings.

The naturalistic approach is more flexible than the other two in the sense that it allows the researcher to analyze the *team mind* on the basis of both team-wide behavior and pair-wise interactions. For this reason, it combines some of the advantages of the collective and holistic approaches and can be readily applied to teams of any size working in real-world settings. The flexibility of this approach comes from its dependence on qualitative analysis of anecdotal data. Although the individual analyses using this approach are rigorous and systematic, the method is not readily transferable and the results not always generalizable.

The structural approach to shared knowledge, proposed in this chapter, combines the advantages of the naturalistic, collective, and holistic approaches. Like the naturalistic approach, it can be applied to teams of any size operating in real-world environments. Its basic unit of analysis is the same type of pair-wise shared mental model used in the collective approach, but it is similar in principle to the holistic approach in the sense that it treats shared knowledge across the team as more than just the sum of its parts. The structural approach is based on the notion that shared knowledge in the team is an emergent property of the team as a whole resulting from individual cognition. Figure 5-1 provides a graphical depiction of each of the approaches using a simple three-person case. The dashed border around the team members in Figure 5-1(a) denotes the observations made by an outside observer following the naturalistic approach, which is not restricted to a particular construct of shared knowledge. The plus signs between team members in Figure 5-1(b) is indicative of the collective view of shared knowledge as the sum of shared mental models in each dyad. Similarly, the multiplication signs in Figure 5-1(c) imply that

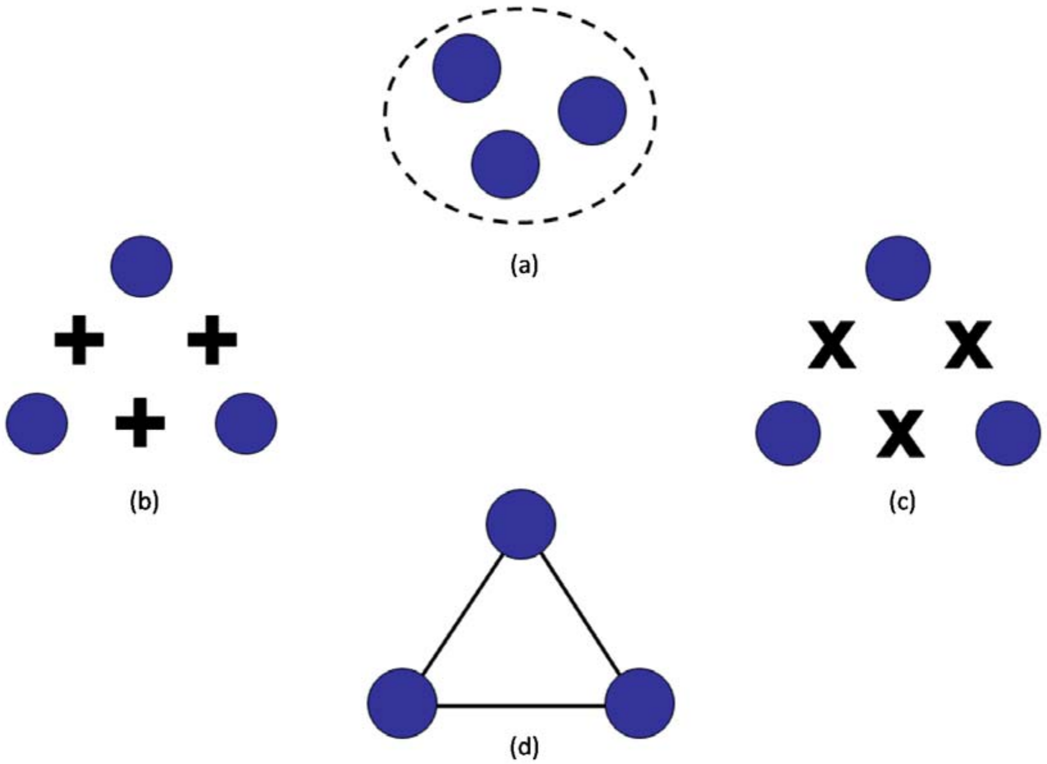


Figure 5-1. Four Approaches to Shared Knowledge. (a) Naturalistic. (b) Collective. Adapted from Cooke and Gorman (2006). **(c) Holistic.** Adapted from Cooke and Gorman (2006). **(d) Structural.**

shared knowledge is more than just the sum of shared mental models of team members but that the interactions themselves dictate the overall knowledge of the team. Finally, the lines between team members in Figure 5-1(d) are normally called edges or arcs of a network, and the circles representing people in this type of depiction are called nodes. This network view of shared knowledge enables the structural approach to shared knowledge. The network-based method of measuring shared knowledge in teams is the focus of the next section.

5.2. Modeling Shared Knowledge in Teams: The Structural Perspective

This section proposes a methodology for modeling shared knowledge in real-world teams using the structural approach. The first subsection briefly introduces network analysis, the methodological approach that forms the basis of the structural approach. The second subsection then develops a metric of mental model "sharedness" (the extent or degree of knowledge sharing) in dyads. Finally, the third subsection discusses a means of filtering out random overlap in mental models through discretization of sharedness values and presents the final network view of the structure of shared knowledge in a team.

5.2.1. A Network Model of Shared Knowledge

Network analysis refers to a set of methods and techniques used to understand global properties among a group of interacting entities. A network consists of nodes that represent entities and edges (or arcs) that connect the nodes according to some type of interaction. A network is said to be directed if the edges have a meaningful direction from one node to the other. Otherwise, the network is undirected. A Design Structure Matrix (DSM), for example, is the adjacency matrix of a directed network. Social networks are those used to analyze communication patterns or relationships among people in organizations (Wasserman and Faust 1999, Newman 2003). In this type of network, a node usually represents a person and an edge a measurable communication or relationship between two people.

Network analysis can be done in a number of ways depending on the precise phenomena of relevance to the system and of interest to the researcher. Just a few of the basic metrics of a network include the number of nodes, n; the number edges, m; the average distance between nodes, l; and the clustering coefficient, C (Newman 2003). Networks can also be analyzed in terms of various measures of centrality (i.e., how "central" certain nodes are relative to others), density of connections, or logical groupings of tightly interconnected nodes (e.g., recall the use of the Newman-Girvan community structure algorithm to cluster the team-based Design Structure Matrix in Chapter 4).

The purpose of studying networks is to understand broad properties related to the overall structure of a system of interacting entities. Thus, network analysis is often also called structural analysis (Wasserman and Faust 1999). In this thesis, such a structural perspective is applied to the measurement of shared knowledge in real-world design teams composed of more than two or three members. This perspective on the problem is called the structural approach because of its emphasis on the structure that emerges from analyzing shared knowledge in this way. In the undirected networks of shared mental models analyzed in this thesis, each node represents a member of the design team, and each edge represents the shared mental model between a pair of team members. The precise distribution of the edges in a network of shared mental models constitutes the *structure of shared knowledge* in the team.

In the DSM presented in Chapter 4, each cell of the matrix – which is equivalent to an edge in a network – could have one of only two values, 0 or 1. Although that is not necessarily the case for all DSMs, it was the most appropriate construction for the ICE environment because the dependencies among parameters are all of different types and are documented at different levels of granularity. In the shared mental model network, on the other hand, the amount of knowledge sharing is measured in the same way for every pair of team members. For this reason, meaningful distinctions among levels of mental model sharedness can be made. As will be discussed later, these different levels determine a weighting scheme for the edges in the network. The next subsection introduces a metric of mental model sharedness in dyads that is later used to determine the weights of the edges in the network.

5.2.2. A Metric of Mental Model Sharedness

In this section, a metric of mental model sharedness among the members of the ICE design team is presented. This metric is based on prior work in the literature on shared mental models, but the precise measure of shared knowledge is necessarily context dependent since it is based on team members' perceptions related to their own particular work. Thus, the measure developed here applies specifically to the space mission design context.

5.2.2.1. Existing Methods of Quantifying Shared Knowledge

As indicated above, the metrics of shared mental models in the literature are always based in some way on the context of the particular team and task under study. Marks et al. (2002), Stout et al. (1999), and Lim and Klein (2006) each asked participants to rate the strengths of the relationships between pairs of items or statements using a Likert scale.[12] In each of these studies, the authors then constructed a network of related items for each participant and used a method called the structural assessment technique to compute a measure of closeness or similarity ranging from 0 to 1 for a dyad. Mathieu et al. (2000) and Mathieu et al. (2005) used a similar technique. Instead of a traditional Likert scale, however, they asked each participant to fill in the cells of an empty matrix with values ranging from -4 to 4 to account for the possibility that two items could be negatively related to each other. They then computed the shared mental

---

[12] Marks et al. (2002) used 1-to-9 scale, whereas the others used a 1-to-7 scale.

model for a dyad by comparing the structural similarity of the two networks. The resulting measure of mental model sharedness ranged from -1 (indicating diametrically opposed views) to 1 (indicating complete sharedness).

Although some of the studies mentioned above examined larger teams, the basic unit of analysis when measuring shared mental models was the dyad. Marks et al. (2000), on the other hand, devised a somewhat different metric to directly quantify mental model sharedness in triads (teams of three). In this case, a similarity score would not suffice. Instead, the authors developed a scoring algorithm in which the contribution to the total score was weighted both by the number of concepts that team members perceived as related and by the number of team members that indicated each of those relationships. For example, one point was assigned if two team members indicated that a given pair of concepts was related, whereas six points were assigned if all three team members agreed that three concepts were interrelated in the same way.

Kameda et al. (1997) took yet a different approach to studying triads. Although they did not use the term "shared mental model" explicitly, they studied a closely related concept that they called cognitive centrality. To measure cognitive centrality, they developed a metric of shared cognition in each pair in the triad. The authors constructed a "belief configuration matrix" in which participants were listed in the rows and beliefs in the columns. In their matrix, a "1" in cell i, j means that person i believes argument j. By multiplying this matrix by its transpose, they obtained an adjacency matrix of a social network in which each cell represented the number of arguments believed in common by the corresponding pair of people. Since they collected data by asking respondents to list all "conceivable arguments" (p. 299), a simple count was sufficient to represent shared cognition.

5.2.2.2. *Measuring Shared Knowledge in Engineering Design*

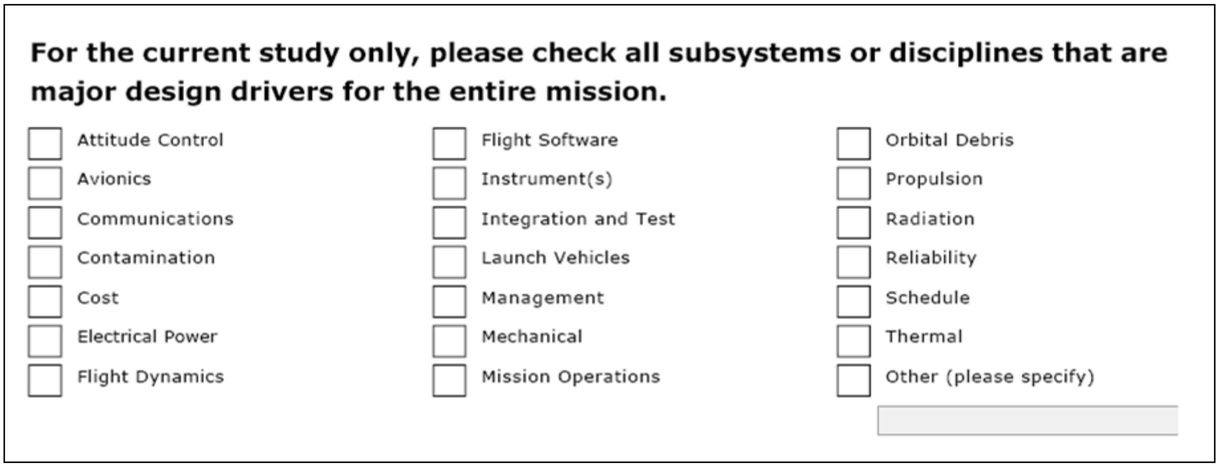
In the real-world, fast-paced concurrent design environment, the measures of mental model sharedness described above could not be used exactly. These measures, which depend on perceived relationships between concepts, are suited to well-defined operational tasks in which all of the individual concepts can be fully articulated. In a creative task such as engineering design, it is not possible to know whether all such concepts represent an exhaustive list because unexpected issues can always arise. Indeed, that is one of the central features of engineering design. Furthermore, because of the rapid pace of the design sessions, it would not have been feasible to ask the engineers to complete pre-session and post-session surveys for the 12 observed design sessions if such a time-consuming question were included. For these two reasons – one theoretical and the other pragmatic – the measure for mental model sharedness used in this research is based on perceptions of individual concepts but not on the relationships among them.

The metric for mental model sharedness proposed here is rather similar to the measure of shared cognition proposed by Kameda et al. (1997). The primary difference is that the survey developed for this research uses a finite list of concepts to measure shared mental models. In the pre-defined task on which their research was based, Kameda et al. (1997) used an open-response survey in which participants were asked to enumerate all possible beliefs regarding the circumstances of a criminal case. In engineering design, however, the complexity of the task implies that the responses given would be too diverse and nuanced to allow for a consistent and comprehensive set of enumerated beliefs. In addition, such a survey question simply would have been too time-consuming to ensure consistently high response rates. Therefore, the survey used in this research asked the members of the team to make a judgment about the major drivers of the design. The survey offered 20 possible choices – the 16 disciplines on the team plus Contamination (formerly a discipline on the team), Instrument(s), Management, and Schedule. An "Other" response was also offered in case any of the team members felt that there were important design drivers for a given session that were not captured among the 20 provided.[13] The question as it appeared on the survey is shown in Figure 5-2.

The data collected from each set of surveys is essentially an *i x j* matrix of people and their beliefs, similar to the one constructed by Kameda et al. (1997). In these surveys, the number of possible beliefs from which to choose was limited by the researchers. Thus, two people could check boxes in common solely because one or both has a personal tendency to check a large number of boxes (or, conversely, they may check none in common because one or both tends to check few boxes). For example, if two people each check 4 out of 10 available boxes with one box checked in common, it would be misleading to characterize their shared mental model based on the commonly checked boxes alone. Therefore, the belief configuration matrix does not

---

[13] Most of the surveys included a total of 20 driver options. On some of the early surveys, fewer drivers were listed. In some cases, one or more respondents used the "Other" option, and this increased the total number of drivers. The total number of drivers for all design sessions was between 19 and 22.



For the current study only, please check all subsystems or disciplines that are major design drivers for the entire mission.

☐ Attitude Control
☐ Avionics
☐ Communications
☐ Contamination
☐ Cost
☐ Electrical Power
☐ Flight Dynamics
☐ Flight Software
☐ Instrument(s)
☐ Integration and Test
☐ Launch Vehicles
☐ Management
☐ Mechanical
☐ Mission Operations
☐ Orbital Debris
☐ Propulsion
☐ Radiation
☐ Reliability
☐ Schedule
☐ Thermal
☐ Other (please specify)

Figure 5-2. Survey Question on Major Design Drivers. The drivers listed are based on the 16 disciplines involved in a typical MDL session plus four other important issues: Contamination (a former discipline on the team), Instrument(s)/Payload, Management, and Schedule. The metric for mental model sharedness is based on common responses to this question.

adequately represent shared mental models when the number of possible beliefs is limited by the available means of data collection.

To account for this issue, the measurement of mental model sharedness in engineering design includes a normalization factor. Whereas simple multiplication of the matrix of responses by its transpose would reveal the number of items checked in common, the metric used here divides the number of common responses by the sum of the number of drivers checked by both people. This metric of mental model sharedness, $S_{x,y}$, is defined as

$$S_{x,y} = 2 \times \left(\frac{D_{x,y}}{D_x + D_y} \right), \tag{5-1}$$

where $D_{x,y}$ is the number of design drivers selected as important in common by team members x and y, D_x is the total number of drivers selected by team member x, and D_y is the total number selected by team member y. The factor of 2 ensures that $S_{x,y} \in [0,1]$.

To demonstrate the 0-to-1 range of possible values, the minimum and maximum possible values of the metric need to be considered. Determining the minimum value is trivial. It is 0 simply because that is the minimum possible value for the numerator, i.e., the smallest number of drivers that can be checked in common.[14] The maximum value is reached when the numerator is maximized and the denominator minimized. The maximum value of the numerator is reached when $D_{x,y} = \min(D_x,D_y)$ because this implies that every driver checked by the person checking the smaller number was also checked by the other person. The denominator is minimized when the larger of D_x and D_y is no larger than the smaller of the two, i.e., when $\min(D_x,D_y) = \max(D_x,D_y)$. This implies that $D_x = D_y$. Given this and the requirement that $D_{x,y} = \min(D_x,D_y)$, the maximum value of the metric occurs when $D_{x,y} = D_x = D_y$. Thus, $\max(S_{x,y}) = 1$, and the full range of possible values of mental model sharedness is $S_{x,y} \in [0,1]$.

Finally, it should be noted that another possible way of measuring sharedness in each dyad in the team would be to compute a simple correlation of their responses to the survey question. According to this method, each person's response would be a series of 20 values – either 0 or 1, indicating that he or she did not or did check a driver, respectively. The correlation of these two strings of 0s and 1s would then be used as the measure of sharedness. This metric has a certain intuitive appeal because it is computed directly from the responses while simultaneously accounting for the tendency of different people to check different total numbers of boxes. In fact, it probably would be a useful metric if the list of choices offered were known to be an exhaustive list of all possible options (as was the case in the work of Kameda et al. 1997). In such a situation, the absence of a check for a given choice can be considered to be as significant as its presence. In the present research, however, each team member might consider the important drivers to be something different from the options listed on the survey (and still choose not to use the Other option). So, given the theoretically unbounded number of possible drivers, the correlation would overvalue the absence of a check by giving it equal importance as the presence of a check. Since this work considers only the presence of a check to be significant, the metric described above is chosen in favor of the correlation.

5.2.3. The Structure of Shared Knowledge: Edge Weights in the Network

With the value of $S_{x,y}$ for each team member x and y on the team established, the next step in modeling shared mental models in the team as a whole is to build a social network to scale the analysis across the entire design team. Before this can be done, though, the sharedness values

---

[14] If no drivers were checked by either person (i.e., the denominator is 0), the value of $S_{x,y}$ is set to 0 because this situation implies that the two team members' mental models are undefined and thus unlikely to be shared.

must be adjusted to account for randomness in the metric. Essentially, the metric as defined above does not in itself capture shared mental models because the precise combination of boxes that a team member checks is subject to some noise based on personal connotations of what the term “major design drivers” actually means. Therefore, this subsection presents a method for filtering the noise from the signal.

Using the survey data, it is not possible to determine a signal-to-noise ratio because there is no empirical or theoretical basis for such a measure in this context. The only way to do this explicitly is to conduct an exhaustive examination of each individual’s understanding of the semantics of the survey question. Given that an in-depth analysis of individual cognition is outside the scope of this research, the alternative used here is an approximation based on the statistics of purely random responses to the survey question.

Consider the possibility that two people answer the survey questions entirely at random. For example, imagine that two hard copies of the survey question are taped to two different walls. Each person throws a random number of darts between 0 and 20 at one of the surveys. After all the darts have been thrown, the two people may have hit some drivers in common by chance, and a value of $S_{x,y}$ as described above could be calculated. The expected value (EV) of $S_{x,y}$ computed in this way is then used as a cutoff value to quantify the shared mental model between the two people.[15] If the value of sharedness for a pair of team members is less than that expected value, it can be said that those two people do not share a mental model to any greater extent than two people with no prior knowledge of the work answering at random.

The expected value is calculated by first enumerating all possible combinations of common driver selections on a pair of surveys. Next, the value of $S_{x,y}$ for each combination is computed. The average of all of those possible values is the expected value. This cutoff alone, however, does not yield a meaningful weighting scale. If every value of sharedness below EV is simply set to 0 while the values above EV remain unchanged, the calculation would overvalue a level of sharedness that is only slightly greater than EV because there would be a discontinuous jump in the scale. Therefore, the entire range of values must be normalized accordingly. To do this, a discrete scale is chosen based on the standard deviation, σ, of the computed expected value. For a survey containing 20 drivers, $EV = 0.444$ and $\sigma = 0.289$. Subject to the constraint implied by

---

[15] For the surveys including more or fewer than 20 drivers, this slight difference was taken into account in the expected value calculation, though the resulting value was similar in all cases.

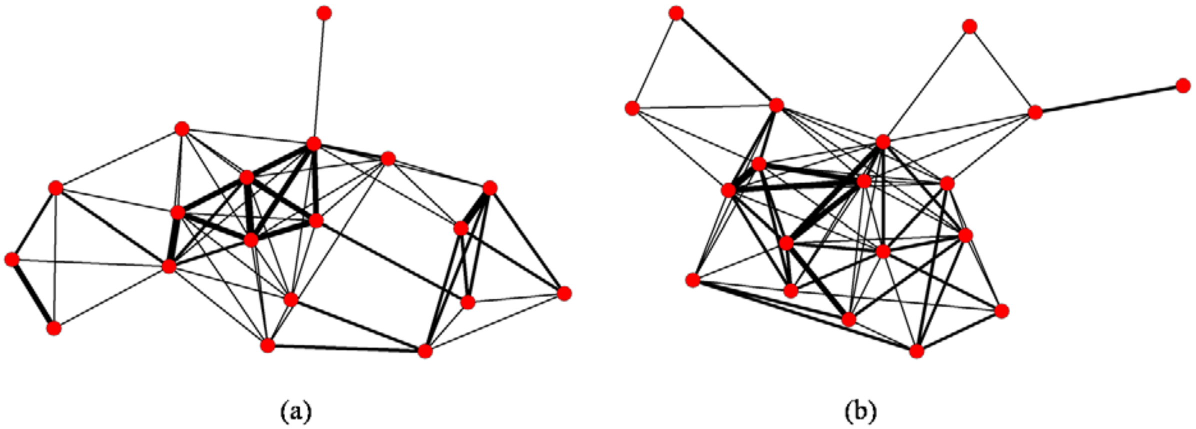


Figure 5-3. Structure of Shared Knowledge for an MDL Session. (a) Pre-Session. (b) Post-Session. The figures represent the structure of shared knowledge for design session 3. Darker edges indicate stronger shared mental models according to a weighting scheme ranging from 0 to 4. The overall structural similarity of these two networks is used to measure the dynamics of shared knowledge in the team.

these values that $EV + 2\sigma > 1$, an integer-valued *shared mental model*, $SMM_{x,y} \in \{0,1,2,3,4\}$, is determined for each pair of team members as follows:

$$\begin{aligned}
SMM_{x,y} &= 0:\ S_{x,y} \leq EV \\
SMM_{x,y} &= 1:\ EV + 0.5\sigma \geq S_{x,y} > EV \\
SMM_{x,y} &= 2:\ EV + \sigma \geq S_{x,y} > EV + 0.5\sigma \cdot \\
SMM_{x,y} &= 3:\ EV + 1.5\sigma \geq S_{x,y} > EV + \sigma \\
SMM_{x,y} &= 4:\ S_{x,y} > EV + 1.5\sigma
\end{aligned} \tag{5-2}$$

While this alleviates the problem of overvaluing levels of sharedness slightly greater than EV to some degree, the problem is not entirely eliminated because there is still some uncertainty at the margins, i.e., for those values of $S_{x,y}$ that are close to one of the cutoff values in the $SMM_{x,y}$ scale. Furthermore, the choice of a 0 to 4 scale is relatively arbitrary. For this reason, section 5.4 presents a sensitivity analysis of the effect of this choice on the outcomes of the analysis.

Once the edge weights have been computed, it is possible to construct the entire social network that represents the shared mental model of the team as a whole. The value of $SMM_{x,y}$ for each x and y is used as the weight of the corresponding edge in the shared mental model network. Figure 5-3 shows two examples of a social network of shared mental models with edges shaded according to this weighting scheme. The network graphs are drawn using the NetDraw software package (Borgatti 2002). The two networks shown represent the pre-session and post-session structure of shared knowledge for one MDL session (session 3 among those observed). These networks are static snapshots of the structure of shared knowledge at different points in time. Separate analysis of these networks could reveal detailed insights about the structure of shared knowledge in the team as it relates to both product and process. Some analysis of this kind was conducted as part of this thesis, but no discernible patterns were measured. Nevertheless, structural analysis of each network is an important area of future research. To facilitate this type of study, Appendix D provides graphical depictions of all 24 shared mental model networks used in this research (a pre-session network and a post-session network for each of the 12 observed design sessions). The remainder of the thesis, however, focuses on the dynamics of shared knowledge as revealed by comparing the structures of the pre- and post-session networks (the raw data are also available by e-mail to avnet@alum.mit.edu). The next section describes the proposed method for analyzing the structure of shared knowledge from a dynamical perspective.

5.3. Measuring the Dynamics of Shared Knowledge

In this thesis, the change in shared knowledge over time is assessed using a measure of structural similarity between the pre-session and post-session network for each design session. The method used to make this comparison is based on the quadratic assignment procedure (QAP). QAP is a statistical method used to determine the overall similarity between two square matrices, A and B, of equal dimension (e.g., two network adjacency matrices). The QAP correlation is a permutation-based procedure that begins with a simple cell-by-cell linear correlation across all non-diagonal cells in A and B.[16] Because the rows and columns of an adjacency matrix are interrelated, the standard significance test for correlations does not suffice. Therefore, QAP determines the significance of the measured correlation by repeating the measurement for A with many permutations of the rows and columns of B. In so doing, QAP preserves the structure of the network as it "explicitly retains the interdependency among the dyads" (Krackhardt 1987, p. 174). The significance is determined by finding the percentage of equivalent random networks that have an equal or greater correlation with network A as does the original network B. In this research, the pre-session matrix is taken to be the expected network,

---

[16] Incidentally, Mathieu et al. (2000) and Mathieu et al. (2005) used the same procedure to measure shared mental models in dyads. Recall from section 5.2.2.1 that they constructed a matrix of related concepts for each person. They then determined mental model sharedness by computing the QAP correlation between the matrices for the two members of each dyad.

A, since it is the starting point, and the post-session network is taken as the measured network, B. The QAP correlation is computed using the UCINET software package (Borgatti et al. 2002). For each design session, 5,000 permutations were tested to determine the significance of the correlation. It should be noted, however, that the metric used from this calculation to measure the dynamics of shared knowledge is the QAP correlation itself and not its significance.

As stated above, the QAP correlation between the pre-session and post-session network for each design session is used to measure the change in shared knowledge over time. Mathematically, however, the metric measures the opposite of that. When comparing the same network at two different points in time, the QAP correlation is a measure of constancy or stability of shared knowledge in the team. Thus, the metric defines a value called *stability of shared knowledge*, C_{SMM}. Since the goal of the present analysis is to measure the *dynamics* of shared knowledge, a simple transformation of C_{SMM} into a measure of change is performed. Accordingly, another metric, called *change in shared knowledge*, is defined as

$$\Delta S = \frac{1 - C_{SMM}}{2}, \tag{5-3}$$

Since $C_{SMM} \in [-1,1]$ by definition, the normalization factor of ½ is applied so that $\Delta S \in [0,1]$.

The value of ΔS indicates the magnitude of change in shared knowledge, but it does not provide any information about the direction of that change. In other words, the metric alone does not specify whether the change is a convergence or a divergence of shared knowledge in the team. To determine directionality, a metric for the team mental model, S, is computed. This metric is simply the average value of $S_{x,y}$ over all team members x and y for each of the shared mental model networks. Thus, the calculation of S is based on the collective approach to shared knowledge. A comparison of this post-session average sharedness, S_{post}, to the pre-session value, S_{pre}, reveals that $S_{post} \geq S_{pre}$ for 11 of the 12 design sessions, indicating that shared knowledge across the team increased over time. In the one session for which $S_{post} < S_{pre}$ (session 9 of the 12 observed), the QAP correlation was actually slightly negative but close to 0. This means that there was no significant relationship between the pre- and post-session networks for that session, so the direction of the change is immaterial. Based on these results, it can be said that ΔS corresponds to an increase in shared knowledge over the course of the design session. From this, it might be concluded that the members of the team *learn from each other* during the work and that ΔS can thus be viewed as a measure of team learning (though the *correctness* of the learned knowledge cannot be determined from the available data). Appendix D provides the actual values of S_{pre}, S_{post}, and ΔS for each of the observed design sessions.

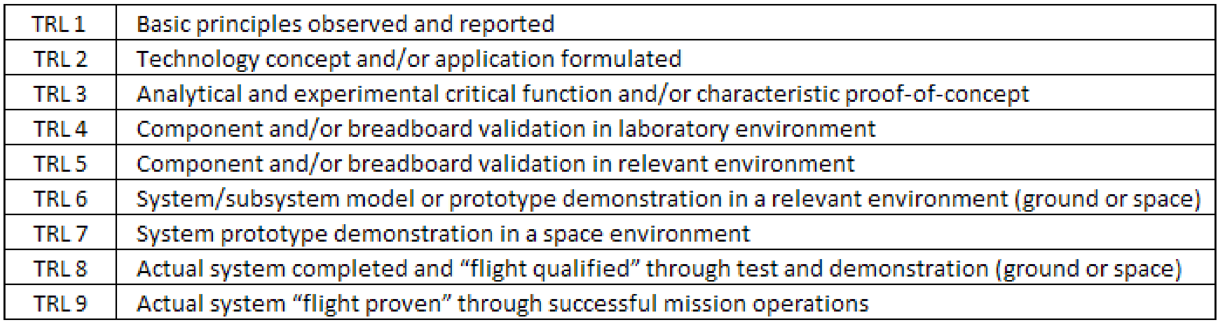
Once the value of ΔS has been computed for each design session, it is necessary to demonstrate that the variations in the value of change in shared knowledge do not simply result from team members' experience answering the survey questions. If the computed value of ΔS were due to respondents' prior knowledge of the survey, some relationship would exist between that metric and the sequence of the design sessions. To test for this, a relationship was computed between ΔS and design session sequence. Because session sequence is an ordinal scale, Spearman's ρ (a rank-based equivalent to correlation for ordinal values) is used to measure this relationship. Among all 12 design sessions, $\rho = 0.280$ and $p = 0.38$. As will be seen in the next section, however, the role of shared knowledge is different for the third-run sessions than for the first-run ones. Considering only first-run sessions, $\rho = 0.552$ and $p = 0.10$. Finally, the three atypical sessions among the 10 first-run ones are, by coincidence, the last three in the sequence. This circumstance could affect the results simply because of the timing of those sessions. For this reason, the relationship is also calculated for just the set of seven first-run typical sessions. In this case, $\rho = -0.179$ and $p = 0.70$. These results indicate that regardless of how the data set is split, no statistically significant relationship exists between ΔS and session sequence. Thus, the survey responses do not result from team members' experience with the survey. Based on this, it is determined that the survey responses are related to the actual content of the design sessions.

Given this finding, the next step is to compare change in shared knowledge to aspects of each session's content. Because ΔS explicitly measures only the change in team members' common views of the important aspects of their work, it cannot be said to imply anything about the team, their process, or the product until it has been compared to system attributes. Therefore, the usefulness of the measure must be tested by comparing it to one or more objective metrics. As such, the next section presents an analysis of ΔS as it relates to various aspects of the technical system being designed.

5.4. Model Evaluation

The purpose of this section is to evaluate the model of shared knowledge based on the structural approach. Each subsection describes a different attribute of the technical system and

Table 5-1. NASA Definitions for the Technology Readiness Level Scale. Adapted from Mankins (1995).



TRL 1	Basic principles observed and reported
TRL 2	Technology concept and/or application formulated
TRL 3	Analytical and experimental critical function and/or characteristic proof-of-concept
TRL 4	Component and/or breadboard validation in laboratory environment
TRL 5	Component and/or breadboard validation in relevant environment
TRL 6	System/subsystem model or prototype demonstration in a relevant environment (ground or space)
TRL 7	System prototype demonstration in a space environment
TRL 8	Actual system completed and “flight qualified” through test and demonstration (ground or space)
TRL 9	Actual system “flight proven” through successful mission operations

demonstrates a relationship between the dynamics of shared knowledge and that metric. These relationships suggest that the metric ΔS provides a useful representation of the dynamics of shared knowledge because it varies with the design concept under study.

5.4.1. Dynamics of Shared Knowledge and Mission Concept Maturity

Technology Readiness Level (TRL) is a discrete scale used by NASA, the Department of Defense (DoD), and some other government agencies to categorize the maturity of a particular technology. Because it is a standardized scale that is applied to variety of technologies, it can be used to make meaningful comparisons of technological maturity across different technologies (Mankins 1995). The TRL scale ranges from 1 to 9, where 1 corresponds to a technology for which basic principles have been observed and 9 to a technology that has actually flown on a mission. Table 5-1 shows the NASA definitions for each of the levels on the TRL scale.

In general, the TRL scale is applied to specific technologies and not to entire mission concepts. Some precedent does exist, however, for scaling the metric to make it applicable to an entire spacecraft, mission, or program. Lee and Thomas (2000) proposed a metric called Weighted average Technology Readiness Level (WTRL), which they defined as the weighted average TRL of all system components weighted by contribution of each component to the total cost. This value is then truncated because TRL is measured on an ordinal scale. Thus, a WTRL of 6.9 would still be considered to be 6 because it has not yet reached the level of 7. In the present research, a system-wide metric called Mission TRL (MTRL) is used to measure overall mission concept maturity. On the post-session survey, each team member (including the Team Lead, Systems Engineer, and members of the customer team) was asked to indicate his or her

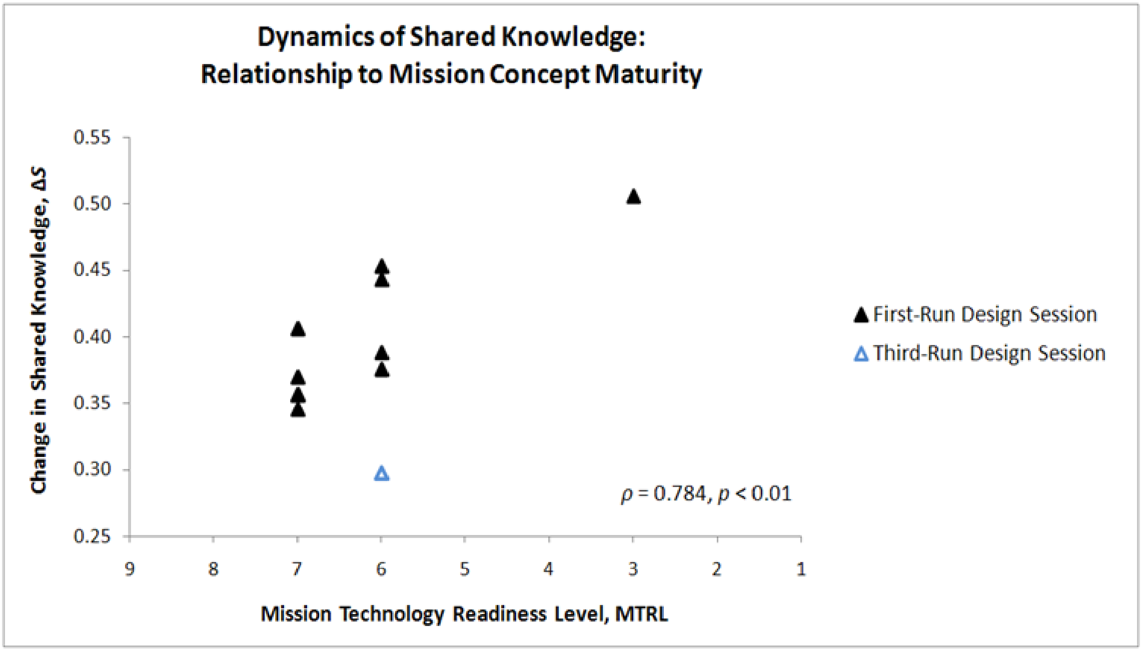


Figure 5-4. Dynamics of Shared Knowledge and Mission Concept Maturity. Because Mission TRL is reported on an ordinal scale, the relationship between the variables is measured using Spearman's ρ. The relationship is shown visually by the location of the points, but a trend line cannot be drawn. The blue-outlined triangles represent third-run design sessions. In these two sessions, shared knowledge does not change as much as would be expected based on the value of MTRL alone.

judgment of an "effective" TRL for the overall system being designed if such a value were to be determined at the mission level. MTRL is defined as the median of all responses provided.

Figure 5-4 demonstrates a relationship between change in shared knowledge, ΔS, and MTRL. Because MTRL is determined on an ordinal scale, it is not possible to draw a trend line or to compute a correlation. Instead, the data points are shown so that the relationship can be seen qualitatively, and a value of Spearman's ρ is reported. In addition, the MTRL scale is shown in descending order. This choice can be made because the direction of the TRL scale is arbitrary and could just as easily be defined in the other direction. The value of ρ is reported as positive, but the sign does not matter for this particular scale because of its arbitrary direction. The positive sign of ρ is merely a result of the display choice and the resulting direction in which ranks are assigned in calculating Spearman's ρ.

As the plot shows, a statistically significant relationship exists between ΔS and MTRL. Thus, over the course of designing a relatively mature mission concept, the team retains a similar level of shared knowledge from beginning to end. Conversely, if the mission concept is less mature, shared knowledge changes more. Intuitively, this result is expected if it is assumed that less mature technologies are less familiar and/or more difficult to design. For systems that are less mature, the individual members of the team can be expected to learn a great deal about the system or change their views about it. For mature system concepts, however, the team would already know at the beginning what they are going to know at the end, so the change in shared knowledge would be less. This interpretation is consistent with the view that members of a new team (or, in this case, a team encountering a new problem) "start out with an abstract, diffuse or general [shared mental] model and specificity increases with experience" (Klimoski and Mohammed 1994, p. 418).

This relationship, however, is sensitive to the design team's particular experience with the concept under study. In the figure, the blue-outlined markers represent MTRL for the two third-run design sessions observed. The low values of ΔS for these two design sessions indicate that this experience has an effect on the dynamics of shared knowledge. For systems with which a large portion of the team has worked previously, shared knowledge changes less than would be expected on the basis of mission concept maturity alone. This implies that the team members' necessarily lower level of familiarity with less mature technologies is mitigated by their past experience with the specific concept being studied. Since only two data points are available for third-run design sessions, however, it is not possible to observe whether a significant relationship exists among those sessions.

Although TRL is a relatively robust metric with broad applicability, it only tells part of the story in terms of testing the metric for change in shared knowledge. First, the version of the metric used here to measure the maturity of the entire mission concept was devised specifically for the purposes of this research. Second, the metric is somewhat subjective. Although the team as a whole is certainly a reliable judge of the overall maturity of the mission concept that they are designing, each person's response to the question is based on his or her own judgment. In addition, because of the ordinal nature of the scale, there are only a few possible data points available. As a result only one of the 12 design sessions has a TRL other than 6 or 7. Thus, it is not possible to make a clear judgment about the relationship between change in shared knowledge and the design product on the basis of this metric alone. To account for these limitations, the following subsections further test the model of shared knowledge by examining the relationship between change in shared knowledge and three distinct objective features of the system.

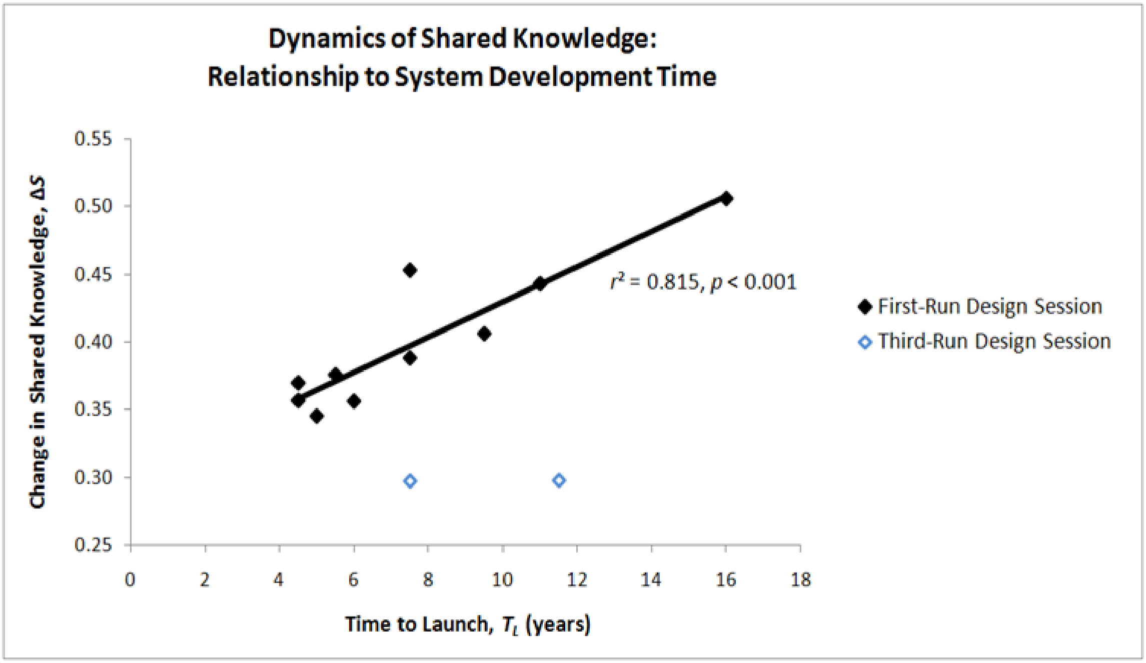


Figure 5-5. Dynamics of Shared Knowledge and System Development Time. Black diamonds represent first-run design sessions, and blue-outlined diamonds represent third-run design sessions.

5.4.2. Dynamics of Shared Knowledge and System Development Time

The typical development time for a scientific spacecraft and mission is on the order of 10 to 20 years. One of the roles of the ICE environment is to expedite the early design process so that concepts can be realized more quickly. Still, even with such a rapid conceptual design completed, the lag time from that point until the launch of the spacecraft is rather long. For complex or advanced missions, the time frame can be even longer. Because of this difference among mission types, the development time for the system can be used as a meaningful technical attribute of the design product. In this research, development time is measured specifically as *time to launch*, T_L, which is the time from the ICE design session to the launch date determined during the session rounded to the nearest half of a year.

Figure 5-5 shows the relationship between time to launch and change in shared knowledge. As the figure demonstrates, a statistically significant correlation exists between ΔS and T_L. This relationship, like the one shown in Figure 5-4, supports a hypothesis that a connection exists between ΔS and technical characteristics of the system. Furthermore, as with MTRL, the two third-run design sessions (the blue-outlined diamonds in the figure), have a lower value for ΔS than would be expected based on T_L alone. Once again, this can be attributed to the team's pre-established familiarity with these otherwise more complex mission concepts. In this case, a trend line can be drawn for the relationship between the two parametric quantities, ΔS and T_L. From the position of the two data points for third-run design sessions, one could imagine a family of such curves based on the number of times that the team has previously worked on the same concept. The curve would essentially be shifted downward along the ΔS axis as the number of past design sessions on the concept increases. Of course, the available data on third-run design sessions are insufficient to demonstrate empirically whether this relationship actually exists.

5.4.3. Dynamics of Shared Knowledge and Launch Mass

The next objective feature of the technical system used to test the model of shared knowledge is *launch mass*, M_L. This metric can mean different things depending on the mission concept considered. For example, a planetary mission is likely to be more massive than a satellite in Earth's orbit. This is true for a number of reasons, including propellant, power needs, and redundancy. In other cases, two missions might be quite similar to one another. In these instances, advanced technologies or innovative approaches could allow the designers to reduce the mass of the system while accomplishing the same objectives. Either way, launch mass is another objective metric that describes certain aspects of the system being designed. For this reason, it is worthwhile to determine whether this metric relates in any way to the dynamics of shared knowledge in the design team.

As shown in Figure 5-6, a statistically significant correlation exists between M_L and ΔS. The figure includes two trend lines. The dotted line is a fit for all 10 first-run data points – the nine solid black diamonds and the one black-outlined diamond. The last point corresponds to a design session of a qualitatively different type from the other observed sessions. In that instance, the customer team did not ask for a full point design. Instead, the session was geared toward concept generation for an advanced mission concept – lunar surface operations. The exact nature of the operations under study cannot be publicly disclosed, but the hardware necessary for any such mission undoubtedly would contribute to an extremely large launch mass relative to the masses for the other sessions. As the dotted trend line shows, a statistically significant correlation exists. Still, the black-outlined data point is an outlier. Given that and the session's difference in type from the others, the relationship is also shown in the figure with that data point removed from the set. The solid line shows the relationship among only the sessions represented

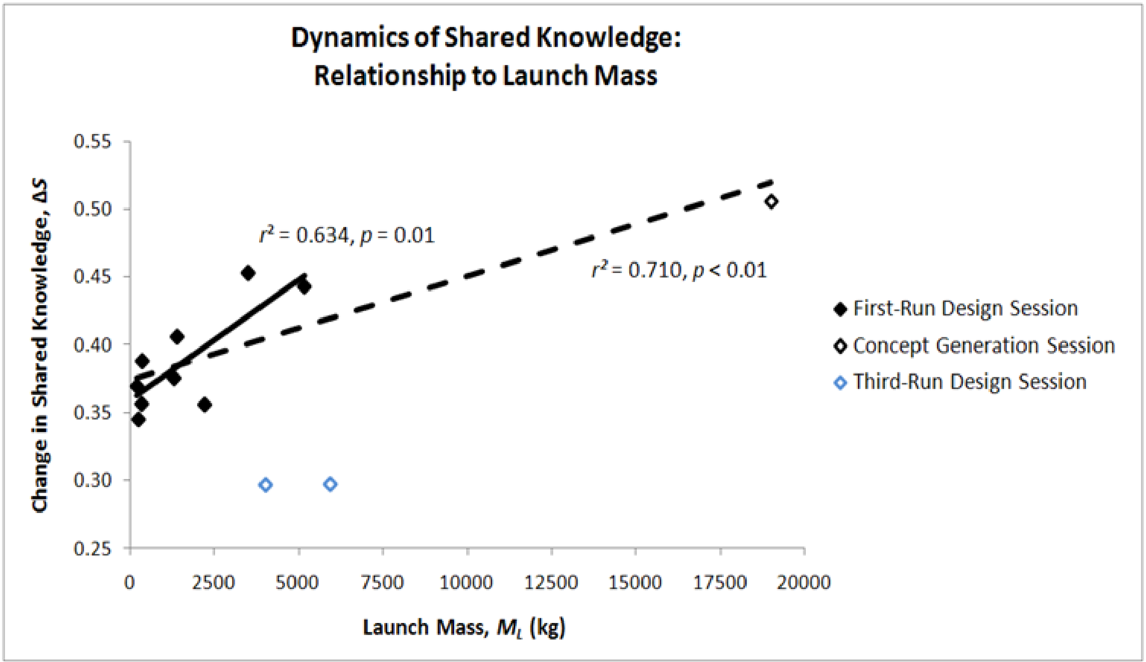


Figure 5-6. Dynamics of Shared Knowledge and Launch Mass. Black diamonds represent first-run design sessions, and blue-outlined diamonds represent third-run design sessions. The black-outlined diamond represents a first-run concept generation session on an advanced mission concept. This data point is an outlier, but the trend exists whether that point is included in the analysis or not.

by the nine solid black diamonds. With just those points, a significant positive linear trend between M_L and ΔS remains, though the slope of the trend line is quite different.[17]

5.4.4. Dynamics of Shared Knowledge and Mission Cost

One of the most important aspects of space systems design is an accurate estimate of the cost of the system. In this research, cost is used as another objective metric of the system because it "is an engineering parameter that varies with physical parameters, technology, and management methods" (Apgar et al. 1999, p. 783). Therefore, this subsection uses mission cost to test the model of shared knowledge in the team.

In space systems design (and in the Mission Design Laboratory in particular), two different types of cost estimates are typically made – parametric and grassroots. Parametric cost modeling uses specific software tools to generate detailed cost estimates of a spacecraft based on

---

[17] With the one extreme point as part of the data set, it also appears as though a curvilinear relationship could exist. Given a large enough sample size, a curvilinear relationship with an asymptotic limit could be expected for all of the plots in this section because of the upper bound at $\Delta S = 1$. In this research, however, the sample size is not large enough to confirm the existence of this type of trend.

a few metrics such as mass and volume. The modeling tools are generally based on experience with past systems. In the MDL, parametric cost is done by a specific discipline engineer whose role is to produce these estimates.[18] A grassroots cost estimate is an aggregate of individual estimates made by the discipline engineers based on actual component and labor costs and, when necessary, the judgment and expertise of the engineer. In the MDL, the Systems Engineer is responsible for maintaining the estimates of the discipline engineers and the total resulting grassroots cost estimate. The parametric and grassroots estimates are considered acceptable for the purposes of conceptual design in the MDL if they are within 40% of each other.

For the purposes of this analysis, the grassroots cost estimate is used. This choice is made for two reasons. First and most importantly, the grassroots result is used to estimate the cost of the entire mission whereas the parametric estimate applies only to the spacecraft hardware.[19] Second, the grassroots estimate is based directly on aggregate results produced by several team members rather than being a product one discipline engineer's independent analysis of the product. For these reasons, the grassroots estimate is likely to be a better indicator of team dynamics and thus a more useful metric against which to evaluate the metric for change in shared knowledge. Of course, both grassroots and parametric could be used here, but this is unnecessary since the two estimates are strongly correlated with each other across sessions. Therefore, the grassroots estimate is used for the purposes of this research.

Figure 5-7 shows the relationship between the grassroots cost estimate, P_{GR}, and ΔS. In addition to the third-run design sessions, which follow the same pattern as with the other metrics discussed above, two other points have been removed from the regression. Session 8, denoted by the green triangle, was primarily a costing exercise and therefore is not considered to be a meaningful data point when comparing sessions according to cost.[20] In addition, session 10 was removed from the data set because the MDL did not calculate a grassroots cost estimate for that design session.

---

[18] In the MDL, Parametric Cost is staffed by the same person as Launch Vehicles for a given session, but the two roles are distinct functions in the design team.

[19] An entire mission cost is then calculated from the parametric estimate using general rule-of-thumb percentages of the spacecraft cost as part of the total mission cost.

[20] Session 11 was intended to be a costing exercise before it began, but the actual work conducted in that session involved a full design. Therefore, session 11 is a valid data point in the trend.

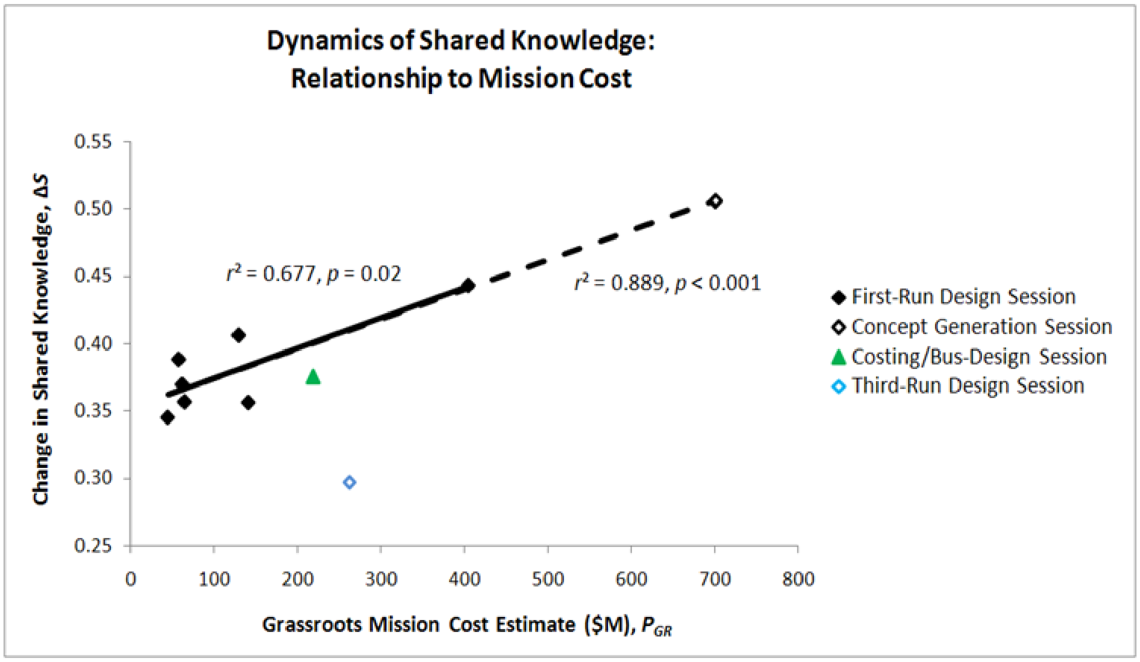


Figure 5-7. Dynamics of Shared Knowledge and Mission Cost. Black diamonds represent first-run sessions, and blue-outlined diamonds represent third-run sessions (the cost for session 1 is beyond the scale shown). The costing session, denoted by a green triangle, is not comparable to other sessions in terms of mission cost. The estimate used for the concept generation session was not determined in the MDL. The figure is an approximate typical cost for similar concepts designed elsewhere. Session 10 does not appear because the MDL did not conduct a grassroots cost estimate for that session, and estimates for similar concepts were not available.

From the eight remaining data points, two trend lines are shown in the figure. The first, denoted by the seven black diamonds and the accompanying black trend line, consists entirely of first-run design sessions using grassroots cost estimates made for each of the design sessions. Although this relationship is statistically significant, six of the seven data points correspond to typical design sessions whose cost estimates do not vary significantly from one mission to the next. As a result, the significance of the trend is dependent on a single available data point for one of the more advanced mission concepts (session 11). To verify that the trend is truly significant and not just a result of the position of that one point, an additional data point was added to the set. The black-outlined diamond represents session 9, the concept generation session. Because that session did not result in a point design, a cost estimate could not be made in the MDL. Fortunately, a comparable estimate was available using a point design created in another context for a similar mission concept.[21] Because of the low maturity of this type of mission concept in general, the estimate is necessarily somewhat uncertain, but the cost is known to be significantly higher than for any of the other designs used in this analysis. Although the trend that includes this data point (the dashed line) cannot be used to establish a relationship between the variables both because of its source and its uncertainty, the value of that point does support an argument that the previously established trend using only MDL grassroots estimates was not spurious.

Based on the testing and evaluation described in this section, it can be said that the model and the metric of change in shared knowledge constitute a useful and meaningful construct for analyzing the social aspects of the space mission design process. Although the results do not imply a causal relationship, they indicate that shared knowledge is connected in some way to the nature of the system being designed. Specifically, the results show that change in shared knowledge is related to mission concept maturity, system development time, launch mass, and system cost. Still, the networks of shared knowledge were constructed with integer-valued edge weights based on a logical but still somewhat arbitrary set of cutoff values. The next section presents a sensitivity analysis to show that the results discussed above are not dependent on those chosen cutoff values.

5.5. Sensitivity Analysis

As discussed previously, the results presented in this chapter are necessarily influenced by certain aspects of the way in which the model is built. In particular, pre- and post-session shared mental model networks are constructed using a discrete set of edge weights, $SMM_{x,y}$, that are assigned according to the values of a continuous metric, $S_{x,y}$. Although the cutoff values are based on statistical properties of the model, the particular choice among the possible cutoffs is at least somewhat arbitrary. Therefore, a sensitivity analysis is necessary to assess whether the results presented above still hold if a different set of cutoff values is chosen.

To do this, the analysis presented above was repeated using three other cutoff values for edge weights in the shared mental model networks. The three schemes chosen are referred to as

---

[21] Because of the MDL customer's need for confidentiality, the source of the cost estimate used cannot be revealed here since that information would also specify the precise nature of the mission concept studied in the MDL.

high-cutoff, low-cutoff, and no-cutoff. In the high-cutoff case, the cutoff values of $S_{x,y}$ are shifted upward by half of a standard deviation. Thus, the values for the high-cutoff case are

$$\begin{aligned} SMM_{x,y} &= 0: \; S_{x,y} \le EV + 0.5\sigma \\ SMM_{x,y} &= 1: \; EV + \sigma \ge S_{x,y} > EV + 0.5\sigma \\ SMM_{x,y} &= 2: \; EV + 1.5\sigma \ge S_{x,y} > EV + \sigma \\ SMM_{x,y} &= 3: \; S_{x,y} > EV + 1.5\sigma \end{aligned} . \qquad (5\text{-}4)$$

As the cutoffs indicate, the scale for $SMM_{x,y}$ in this case ranges from 0 to 3 and errs on the side of filtering out noise at the possible expense of signal. Similarly, in the low-cutoff case, the cutoff values of $S_{x,y}$ are shifted downward by half of a standard deviation. In the low-cutoff case,

$$\begin{aligned} SMM_{x,y} &= 0: \; S_{x,y} \le EV - 0.5\sigma \\ SMM_{x,y} &= 1: \; EV \ge S_{x,y} > EV - 0.5\sigma \\ SMM_{x,y} &= 2: \; EV + 0.5\sigma \ge S_{x,y} > EV \\ SMM_{x,y} &= 3: \; EV + \sigma \ge S_{x,y} > EV + 0.5\sigma \\ SMM_{x,y} &= 4: \; EV + 1.5\sigma \ge S_{x,y} > EV + \sigma \\ SMM_{x,y} &= 5: \; S_{x,y} > EV + 1.5\sigma \end{aligned} . \qquad (5\text{-}5)$$

Therefore, the scale for $SMM_{x,y}$ in this case ranges from 0 to 5 and errs on the side of including as much signal as possible at the expense of including additional noise. Finally, in the no-cutoff case, the transformation is simply

$$SMM_{x,y} = S_{x,y} . \qquad (5\text{-}6)$$

In this case, no filtering is applied, so all signal and all noise are included in the analysis.

Figure 5-8 shows the sensitivity of the metric of change in shared knowledge, ΔS, to the cutoff value chosen. As the figure indicates, a strong positive correlation exists between ΔS as measured using the original cutoff and ΔS as measured using each of the three alternative cutoffs. Thus, the value of change in shared knowledge for each design session is not significantly affected by the exact cutoff values chosen. This result is important for establishing the strength of the model. Still, it does not necessarily imply that the relationships between ΔS and the technical features of the system remain unaffected.

To examine the actual impact of the chosen cutoff value on the relationship between ΔS and the system being designed, the full analysis presented in section 5.4 was repeated using the

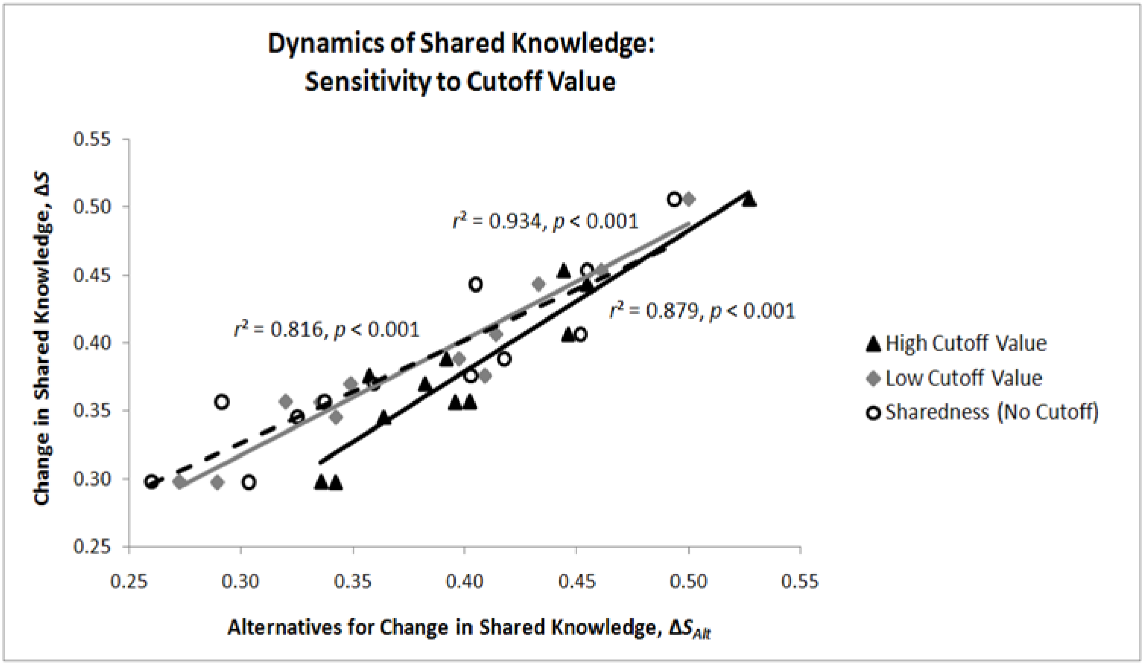
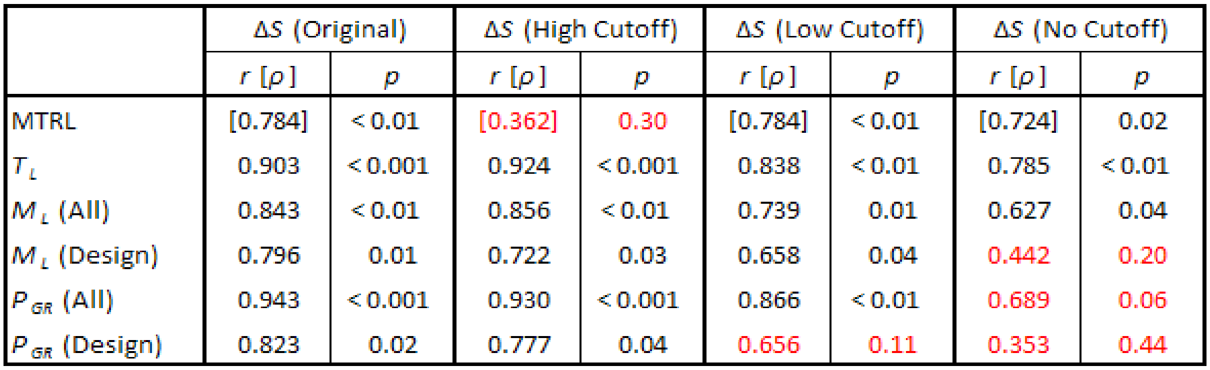


Figure 5-8. Sensitivity Analysis of Sharedness Cutoff Values. This plot demonstrates that the computed values for change in shared knowledge, ΔS, are not significantly dependent on the cutoff value chosen. Both axes of this plot contain a measure of change in shared knowledge. The vertical axis represents the values of ΔS as computed using the original cutoff scheme, ΔS (Original). The horizontal axis represents the three results for ΔS using the alternative schemes: ΔS (High Cutoff) , ΔS (Low Cutoff) , ΔS (No Cutoff). The three trend lines show a strong correlation between ΔS as computed using the original cutoff scheme and ΔS as computed using the alternative cutoff schemes. Thus, the values of ΔS are largely independent of the precise cutoff chosen.

values of ΔS as computed based on all three alternative cutoff schemes. The results of this analysis are presented in Table 5-2. The correlations that are computed using Spearman's ρ to account for the ordinal scale of MTRL are shown in square brackets. As the table indicates, the overall results are not materially affected by the choice of the cutoff value. In the high-cutoff case, all correlations except one remain statistically significant. The value of ρ for the relationship between MTRL and change in shared knowledge is not statistically significant when the high cutoff is used. In the low-cutoff case, the situation is similar. All correlations except one remain significant. The exception in this case is the relationship involving mission cost, and this occurs only when the concept generation session is not included in the analysis. When that session is included, the significance is retained. In the no-cutoff case, three of the six correlations lose statistical significance. It is important to note, however, that this is the base case that contains the maximum amount of noise. For this reason, it is expected that the

Table 5-2. Sensitivity of Relationships to Sharedness Cutoff Values. The technical features listed here are Mission TRL (MTRL), time to launch (T_L), launch mass (M_L), and grassroots cost (P_{GR}). The correlations for M_L and P_{GR} with and without the inclusion of the advanced concept generation session are denoted by (All) and (Design), respectively. Correlations in square brackets are computed using Spearman's ρ because MTRL is measured on an ordinal scale. Red font indicates those correlations that are not statistically significant to at least the $p \leq 0.05$ level.



	ΔS (Original)		ΔS (High Cutoff)		ΔS (Low Cutoff)		ΔS (No Cutoff)	
	r [ρ]	p	r [ρ]	p	r [ρ]	p	r [ρ]	p
MTRL	[0.784]	< 0.01	[0.362]	0.30	[0.784]	< 0.01	[0.724]	0.02
T_L	0.903	< 0.001	0.924	< 0.001	0.838	< 0.01	0.785	< 0.01
M_L (All)	0.843	< 0.01	0.856	< 0.01	0.739	0.01	0.627	0.04
M_L (Design)	0.796	0.01	0.722	0.03	0.658	0.04	0.442	0.20
P_{GR} (All)	0.943	< 0.001	0.930	< 0.001	0.866	< 0.01	0.689	0.06
P_{GR} (Design)	0.823	0.02	0.777	0.04	0.656	0.11	0.353	0.44

least significant correlations would appear in this case. The originally chosen cutoff, on the other hand, yields significant results for all six correlations presented.

This sensitivity analysis has a few important implications for the strength of the model. First, the results demonstrate that the model is generally robust to changes in the cutoff and thus are not merely an artifact of the particular cutoff value chosen. At the same time, however, the reduced significance of the results in the no-cutoff case supports the assumption that a cutoff was necessary, and the slightly lower overall significance of the results in the high- and low-cutoff cases even provide some support that the expected value, as chosen originally, is the best cutoff among those tested to reduce noise while minimizing loss of the signal. Therefore, the sensitivity analysis demonstrates that the model as originally constructed provides a meaningful representation of the dynamics of shared knowledge in the team.

5.6. Discussion: Shared Knowledge and the Design Process

This chapter has presented a model of the dynamics of shared knowledge in engineering design teams. But what does shared knowledge actually mean? As it is defined in this research, it specifically refers to the extent to which team members agree on the major design drivers for a given session. The analysis has shown that the change in these common views over time is related in some way to various technical features of the system being designed. Clearly, this insight has implications for the way that the design session is conducted, but the exact nature of those implications depends on the specifics of the design process.

In Chapter 4, the ICE design process was analyzed using a system-wide representation of information flow. Chapter 5 has developed a model of shared knowledge in the team. Presumably, information flow in the design process must be related in some way to shared knowledge among the members of the team. Therefore, the remainder of the thesis will consider these two aspects of the analysis together by exploring the linkages between the technical and the social in the design process.

The next step in the analysis is to explicitly connect the model of shared knowledge to technical aspects of the product and the process. The discussion addresses a number of important questions that have remained open until now. What do the correlations between change in shared knowledge and the technical features of the system mean for process and product? Does the content of shared mental models (i.e., the actual boxes that the team checks and not just whether they checked them in common) matter? Does shared knowledge relate in any way to the work of particular subsystems and disciplines involved in the design? What is the relationship between the Design Structure Matrix representation of the design process and the dynamics of shared knowledge in the team?

The purpose of the next chapter is to address each of these questions about the connection between design process and shared knowledge in the team. Based on that integrated analysis, a relationship is demonstrated among shared knowledge, team coordination in the design process, and the product of the design.

[This page intentionally left blank]

Chapter 6

Thinking and Doing: Shared Knowledge in the Space Mission Design Process

In Chapters 4 and 5, two distinct models of the design process in the Integrated Concurrent Engineering (ICE) environment were presented. Chapter 4 demonstrated a means of analyzing the process from a technical perspective by managing information flow using the Design Structure Matrix (DSM). Chapter 5, on the other hand, developed a methodology for modeling the dynamics of shared knowledge in the design team. In this chapter, those two models – one analyzing process and the other analyzing people – are integrated. The goal of this chapter is to demonstrate the relationship between information flow in the design process and shared knowledge in the team. In the first section, the results presented in Chapter 5 are collected and reorganized to demonstrate an overall relationship between shared knowledge and the product of the design. The second section discusses the actual content of shared mental models (SMMs) in the space mission design context. Based on this and the results of the technical design process analysis from Chapter 4, the third section argues that the maturity of the Communications subsystem is an indicator of shared knowledge in the design process. Then, the fourth subsection presents an analysis of the connection between team coordination and shared knowledge. The fifth section integrates the results of the first and fourth sections to explore the relationship among shared knowledge, team coordination, and the product of the design. Finally, the sixth section concludes the analysis portion of the thesis and introduces the discussion of recommendations and implications offered in the last two chapters.

6.1. Dynamics of Shared Knowledge and the Design Product

In the previous chapter, a model of shared knowledge in the design team was presented, and the relationship between change in shared knowledge over time and certain technical attributes of the system were used to test the model. This section presents those same results again but looks at them from a somewhat different perspective. The plots that demonstrate all of these relationships are shown together in Figure 6-1. Collectively, they paint a picture about the relationship between the product of the design and shared knowledge in the team. But what is the nature of that relationship?

Using more general product design terminology, the four metrics describing the system are concept maturity, development time, size, and cost. Intuitively, these metrics are related to each other in some way. Table 6-1 shows the relationships that exist among these parameters. According to the results presented in the table, all of the metrics analyzed are interrelated over the 12 design sessions observed except that launch mass and mission cost are not directly correlated with each other. This implies at least the possibility that these attributes are all related to a single overarching property of the system. From this point forward, the potential “umbrella” metric related to the other technical attributes will be referred to as complexity. In general, the term complexity has been defined in a number of different ways, many of which use direct quantitative measures of the system itself. For the purposes of this research, however, complexity is defined merely as the intangible attribute of space systems that may be responsible for the cross-correlations among concept maturity, development time, size, and cost. Using this definition of complexity, the results shown in Figure 6-1 indicate that complexity is positively correlated with change in shared knowledge. Thus, shared knowledge in the team converges more over time (i.e., the team learns more) during the design of more complex systems.

Based on this result, it can be said that for less complex mission concepts, the team already knows at the beginning what they are going to know at the end, i.e., they do not learn a great deal by doing the design work. Therefore, they can concentrate their effort on completing multiple design iterations and/or improving the quality of the design with each iteration. If, on the other hand, the object of a design session is more complex, one can predict based on the results in Figure 6-1 that shared knowledge is likely to converge more. The observed difference in team learning (i.e., change in shared knowledge) during complex design tasks and during less complex ones illustrates the concept of exploration versus exploitation. As March (1991) notes, “maintaining an appropriate balance between exploration and exploitation is a primary factor in system survival and prosperity” (p. 71). A certain amount of planning prior to the start of design sessions on high-complexity mission concepts could help the team to strike this balance by improving their knowledge on complex concepts before the work begins. Conversely, planning

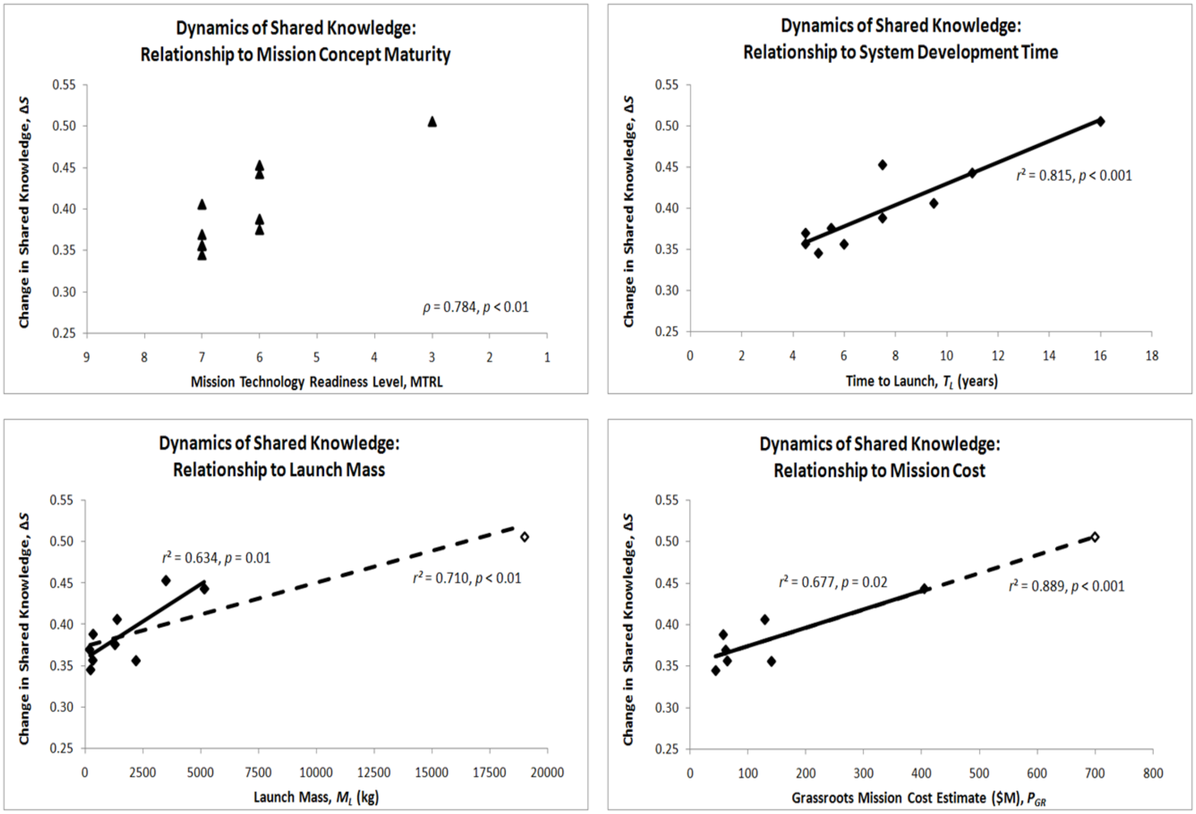


Figure 6-1. Dynamics of Shared Knowledge and the Design Product. The correlations are indicative of a connection between shared knowledge and the system under study. The x-axes of the four graphs are mission concept maturity, system development time, launch mass, and mission cost. These metrics might be considered as separate implications of another system property that is referred to here as complexity. To the extent that a change in shared knowledge implies learning in the team, the four graphs together indicate that team learning increases most during the design of particularly complex systems. (Note that the third-run design sessions shown in the equivalent graphs in Chapter 5 have been removed here for clarity.)

Table 6-1. Correlations among Product Attributes. According to these correlations, MTRL, T_L, M_L, and P_{GR} are all mutually correlated except that launch mass and mission cost are not directly correlated with each other. Both of them, however, are correlated with MTRL and T_L. All correlations involving MTRL are calculated using Spearman's ρ.

	MTRL		T_L		M_L		P_{GR}	
	ρ	p	r	p	r	p	r	p
MTRL	-	-	-	-	-	-	-	-
T_L	0.691	0.01	-	-	-	-	-	-
M_L	0.728	< 0.01	0.883	< 0.001	-	-	-	-
P_{GR}	0.673	0.02	0.649	0.03	0.527	0.10	-	-

for the design of less complex concepts could also contribute to that balance by helping the design team members to expand their thinking about an otherwise relatively mundane task.

The existing literature on teams lends further support to the importance of planning, especially for design sessions involving more complex concepts. Some have argued that pre-performance planning is a crucial step for teams. The reason, however, is not that it provides an immutable and efficient step-by-step procedure to follow. On the contrary, in a dynamic and uncertain environment, plans are likely to change over time. Still, the basic rationale for planning is that the plan influences the way people think about the task (Klein and Miller 1999). It is a valuable activity "just for what is learned by considering alternative actions and learning what will work and what will not" (p. 204). Empirical research on the subject has shown that teams that engage in planning activities prior to periods of high work load developed better shared mental models and demonstrated a higher level of performance (Stout et al. 1999).

This planning, of course, does not have to be limited to the highest complexity concepts. First, nearly all ICE design sessions can be considered by definition to be periods of high workload, and a more complex mission concept does not necessarily imply more difficult work (though it often does). Second, planning could mitigate the potential negative effects of groupthink. The observation that shared knowledge changes less for lower complexity missions implies that the team starts with a certain level of shared knowledge and retains that level when the session ends. This, however, does not imply that this shared knowledge is the *right* knowledge. In fact, a lack of change in shared knowledge could lead to adverse outcomes if the knowledge is flawed. This retention of incorrect knowledge throughout the team is essentially an example of groupthink. Furthermore, it would be difficult in these cases to know what the correct knowledge is until after the design session has been completed. Only by exploring the content of the shared mental models can this issue of the right versus the wrong shared knowledge be considered. Thus, the next section presents an analysis of the content of shared mental models in the design team.

6.2. The Content of Shared Mental Models

Until now, the discussion of shared mental models has focused entirely on the extent to which team members share them. The purpose of this section is to explore the actual content of that shared knowledge and the implications of the content. The first subsection discusses the type of content in shared mental models as they are examined in this research. The second subsection discusses ways of measuring the content of SMMs and the particular role of the Communications subsystem that this analysis reveals.

6.2.1. What Type of Shared Knowledge Do Major Design Drivers Measure?

As discussed in the review of the literature on shared mental models presented in Chapter 2, shared knowledge generally is categorized according to two types of knowledge: task-based and team-based. Task-based knowledge refers to a team member's understanding of the facts, figures, and procedures relevant to completing the specific tasks that need to be performed. Team-based knowledge, on the other hand, is a team member's knowledge of the other members of the team, including their knowledge and capabilities, how they work, and their interactions with the rest of the team. Task- and team-based shared mental models capture the extent to which members of the team hold these two types of knowledge in common. Most of the extant literature on shared mental models either addresses both types of knowledge separately or explicitly notes which one is considered in a given study.

In that literature, the observed teams generally performed well-specified operational tasks. For engineering design, however, the distinction among types of shared knowledge must be more nuanced since the work is more ambiguous and even more subjective. Nevertheless, the present research should be no exception to the convention of clearly specifying the type of shared knowledge being analyzed, as it has important implications for the content of the mental model and therefore for the meaning of the results.

Badke-Schaub et al. (2007) recognized the need for finer distinctions among the types of mental models employed in engineering design teams and so proposed three additional types of knowledge specific to this kind of work. These three types of knowledge are based on process, context, and competence. A process-based mental model refers to a team member's knowledge about how the work is done. A context-based mental model includes knowledge about the organization, the customer, the market, and perhaps the relevant regulatory environment. Finally, a competence-based model involves a person's perceptions of the capabilities of the team as whole.

As these definitions imply, the five types of mental models (task, team, process, context, and competence) are not necessarily mutually exclusive. For example, a process-based model must contain elements of both task- and team-based mental models since it is meant to capture how the people on the team go about completing the task. Similarly, a competence-based model can be seen as an aggregated form of a team-based mental model with some aspects of the context-based mental model included. Because of these overlaps among the types of shared knowledge, it would not be possible for any but the most straightforward design task to operationalize one type of shared mental model unambiguously. This is certainly true in the case of the present research. Shared mental models were measured here using common views of design drivers, which do not directly represent any of the five types of mental models mentioned above. Still, it is informative to explain the aspects of each type of mental model that are captured in the measure proposed in this thesis.

Among the five types of shared mental models, the metric used in this research is most similar to the task-based mental model. The reason for this is that the question on major design drivers asks team members to specify the aspects of the system that drive the specific task of designing the spacecraft and surrounding mission architecture for a given design session. Still, in some ways, the metric also contains features of a team-based mental model. Because 16 of the 20 possible design drivers map directly to a specific member (or, occasionally, two members) of the team, the question could easily be interpreted as asking about the person filling each of those roles and not about the features of the design. Indeed, in some ways, these two factors – task and team – are inseparable in the multidisciplinary ICE environment.

Furthermore, this metric of shared knowledge contains aspects of the competence-based mental model since it is geared specifically toward the issues that drive the design as conducted in the Mission Design Laboratory. Therefore, when answering the question, the team members automatically consider the MDL process and its relative strengths and weaknesses as a design center when answering the question. For example, a typical design session in the MDL is an Earth-orbiting single-spacecraft mission with Earth or space science objectives. This means that team members' judgments of what constitutes a major design driver are necessarily based, at least in part, on the particular competencies that the team has developed in designing this particular type of space mission.

From a process standpoint, the MDL involves the customer team directly in the entire process, but many other ICE design centers do not. Therefore, the MDL might be expected to have a higher tolerance for ambiguity in design requirements since the customer can clarify such issues as they arise. Whereas a poorly defined mission requirement might be seen as a major design driver by other ICE teams, that same requirement might not be as serious an issue in the MDL due to the team's immediate access to the customer. For this reason, the selected design drivers could be dictated in part by the specific design process of the particular ICE center.

Finally, the network model of shared knowledge developed for this research explicitly incorporates a context-based element into the calculation. This happens because the views of the customer team (explicitly made a part of the MDL process) are included in the network of shared mental models. This means that the context (specifically the customer) is part of the emergent structure of shared knowledge across the team even if it is not directly included in the mental models of the individuals. Furthermore, the issues of process and competence discussed above are related to the organizational context of the ICE center, so some element of context is captured in individual mental models as well, albeit indirectly.

Although the subtleties in these distinctions complicate the categorization of the shared mental models measured in this research, they also provide important insight to guide the interpretation of the results. When comparing this research to prior work on shared knowledge in teams performing operational tasks, it might be best to categorize the shared mental models used here as task-based or perhaps even as a hybrid type of mental model. When considering the direct implications for engineering design, however, it is most reasonable to simply categorize the shared mental model based on what it specifically addresses. Thus, this research might be said to measure *driver-based* shared mental models.

Regardless of the term used to describe the type of knowledge, the method for measuring shared knowledge is the primary determinant of the *content* of mental models. The meaning of the results can be accurately discerned only when judged with full consideration of what is actually contained in a shared mental model. Therefore, the next subsection considers the relationship between the content of shared mental models and the nature of the system being designed.

6.2.2. Measuring the Content of Shared Mental Models

In Chapter 5, a network model of shared knowledge in the team was presented, but the model did not account directly for the content of the shared knowledge. It simply measured the common responses from the team members without any direct consideration of what the responses chosen in common were. This section, on the other hand, proposes a complementary metric that captures the content of shared mental models in the team.

To understand the metric for shared knowledge content, consider an *M x N* matrix, *D*, that shows *M* team members down the rows and *N* design drivers across the columns. The contents of the matrix are binary – 0 or 1. A "1" in cell *i,j* indicates that person *i* checked driver *j*, and a "0" in that cell indicates that person *i* did not check driver *j*. The network model of shared knowledge proposed in Chapter 5 focuses on common responses across each pair of rows without regard for the headings of the columns. Conversely, the metric proposed here is concerned with the totals of the columns and does not consider the labels of the rows.

This metric for the content of shared knowledge in the team is simply the proportion of all team members that indicated whether each of the possible design drivers is a major one for the current system. The *perceived importance of design driver j*, $I_{P,j}$, is defined as

$$I_{P,j} = \frac{\sum_{i=1}^{M} D[i, j]}{M}, \tag{6-1}$$

where *D* is an *M x N* matrix of all design driver survey responses, *M* is the number of members of the team for which survey data were collected, *i* is an integer index representing each successive member of the team, and *j* is an enumerated alphabetic index (e.g., Comm for Communications or Prop for Propulsion) representing the particular design driver for which the metric is calculated.

Unlike change in shared knowledge, ΔS, which is an aggregate of the entire team's shared knowledge over time, this metric is intended to represent the perceived importance of each possible design driver so that the contribution of each to shared knowledge in the team can be determined. Therefore, $I_{P,j}$ is different both for pre- and post-session data and for each discipline. Thus, for each design session, there exist up to 40 values of $I_{P,j}$, two (pre-session and post-session) for each of the 20 possible design drivers.

As described in Chapter 5, each subsystem engineer was asked on the post-session surveys to rate the technological maturity of his or her subsystem using the Technology Readiness (TRL) scale. Based on these data, the analysis included a test to determine whether a relationship exists between the technological maturity of each possible design driver and its perceived importance in the team. This relationship was calculated by measuring six correlations (or as many of them that were applicable) for each driver. These correlations were the ones that can be measured among four variables for each possible driver, j: TRL of j, $I_{P,j\ (Pre)}$ (pre-session perceived importance of j), $I_{P,j\ (Post)}$ (post-session perceived importance of j), and ΔS (change in shared knowledge in the team). As with MTRL, all correlations involving TRL are computed using Spearman's ρ. For 19 of the 20 possible design drivers, one-third or fewer of the measured correlations were statistically significant to a $p \leq 0.05$ level. For the one remaining driver, however, all six of the correlations were significant. That one driver was the Communications subsystem. See Appendix E for the full analysis showing all computed correlations for each of the 20 possible design drivers.

Because Communications was the only discipline for which a significant relationship was observed in all cases, the discussion here will focus on that subsystem. The first step in the analysis for Communications was to measure the correlation between subsystem TRL and each the pre- and post-session perceived importance of Communications, $I_{P,Comm}$. For subsystem TRL and pre-session perceived importance, the result is $\rho = 0.688$ and $p = 0.03$. For subsystem TRL and post-session perceived importance, the result is $\rho = 0.636$ and $p = 0.05$. These correlations indicate that the team considers Communications to be a major design driver when its technological maturity is low. This suggests that the team has a better understanding of Communications than they do of any of the other subsystems. Presumably, a subsystem for which the technology is immature will be more difficult to design than one for which the technology is established and proven, but the team as a whole seems to be aware of the specifics of this only for the Communications subsystem. Since the team members rated the importance of Communications across the design sessions accordingly, it appears that they had a strong understanding of some feature (or features) of this subsystem that they did not recognize as readily for other disciplines. To explore this phenomenon further, the next section explains the relationships between aspects of the Communications subsystem and the dynamics of shared knowledge in the team.

6.3. The Communications Subsystem: An Indicator Discipline?

In the previous section, a relationship was demonstrated between the content of shared mental models in the design team and certain technical aspects of the system. For Communications, the relationship implies that the team as a whole has a relatively rich understanding of that subsystem. Thus, it is possible that Communications plays an important role in the overall design process. The purpose of this section is to explore this possibility in greater depth both from the perspective of shared knowledge as discussed in Chapter 5 and in terms of information flow in the design process as described in Chapter 4.

6.3.1. Dynamics of Shared Knowledge and the Communications Subsystem

Section 6.1 provided a summary of the results connecting change in shared knowledge, ΔS, to various aspects of the product of the design. Among the metrics used was Mission Technology Readiness Level (MTRL), a measure of overall mission concept maturity. Given the observed correlations between ΔS and these system attributes, it is also useful to understand whether such a relationship also exists between ΔS and the TRL of each individual subsystem. Thus, the TRLs of the spacecraft subsystems were compared directly to change in shared knowledge to determine whether any subsystem's technological maturity is related to the dynamics of shared knowledge in the team.

Figure 6-2 demonstrates a statistically significant relationship between Communications TRL and ΔS using Spearman's ρ. This result means that shared knowledge changes most when the technology used in the design of the Communications subsystem is less mature. Thus, it can be said that Communications TRL is an indicator of shared knowledge in the team. Now recall from the previous section that Communications TRL is related to the perceived importance of Communications in the team. Since the TRL for the subsystem is related both to ΔS and to $I_{P,Comm}$, it is likely that these two metrics are related to each other as well. Figure 6-3 verifies that this relationship exists, and the trends shown in that figure demonstrate that the perceived importance of Communications is indicative of change in shared knowledge in the team.

As these results indicate, change in shared knowledge, the importance of Communications in that shared knowledge, and the maturity of the Communications subsystem are all mutually correlated. Thus, the Communications subsystem can be viewed as an indicator of the dynamics of shared knowledge in the design team, though it is not immediately apparent what the

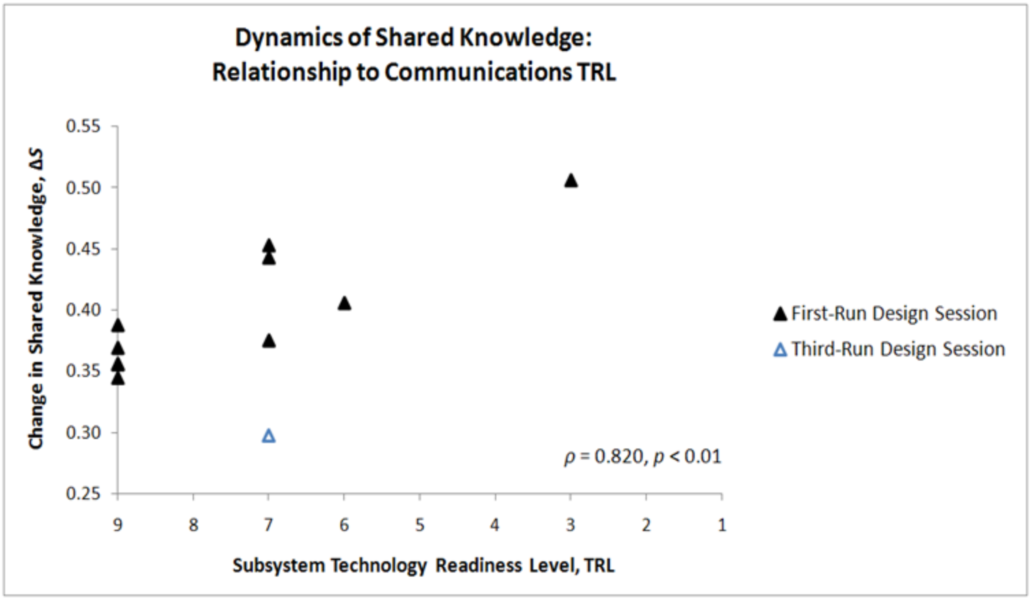


Figure 6-2. Dynamics of Shared Knowledge and Communications Subsystem Maturity. The TRL of the Communications subsystem appears to serve as an indicator of shared knowledge in the team. This is the only subsystem for which such a relationship exists. Black triangles represent first-run design sessions, and the blue-outlined triangle represents a third-run session (the Communications engineer was not involved in one third-run design, session 14).

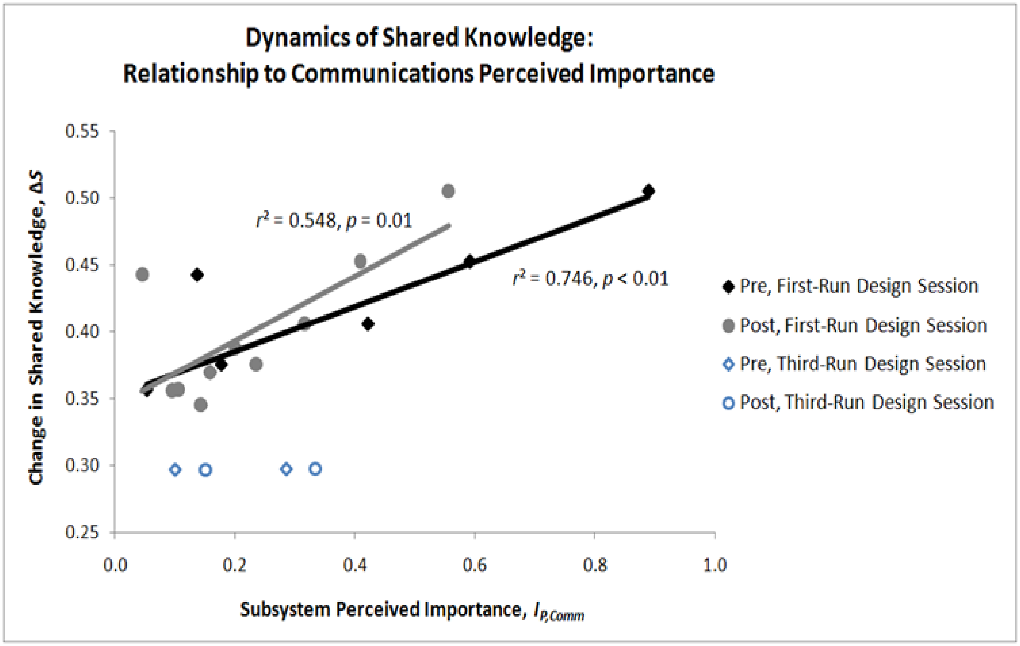


Figure 6-3. Dynamics of Shared Knowledge and Perceived Importance of Communications. The perceived importance of the Communications subsystem appears to serve as an indicator of shared knowledge in the team. This is the only subsystem for which this trend exists. Black diamonds represent pre-session perceived importance for first-run design sessions, and gray circles represent post-session perceived importance for the same sessions. Blue-outlined diamonds and circles correspond to pre-session and post-session perceived importance, respectively, for third-run sessions.

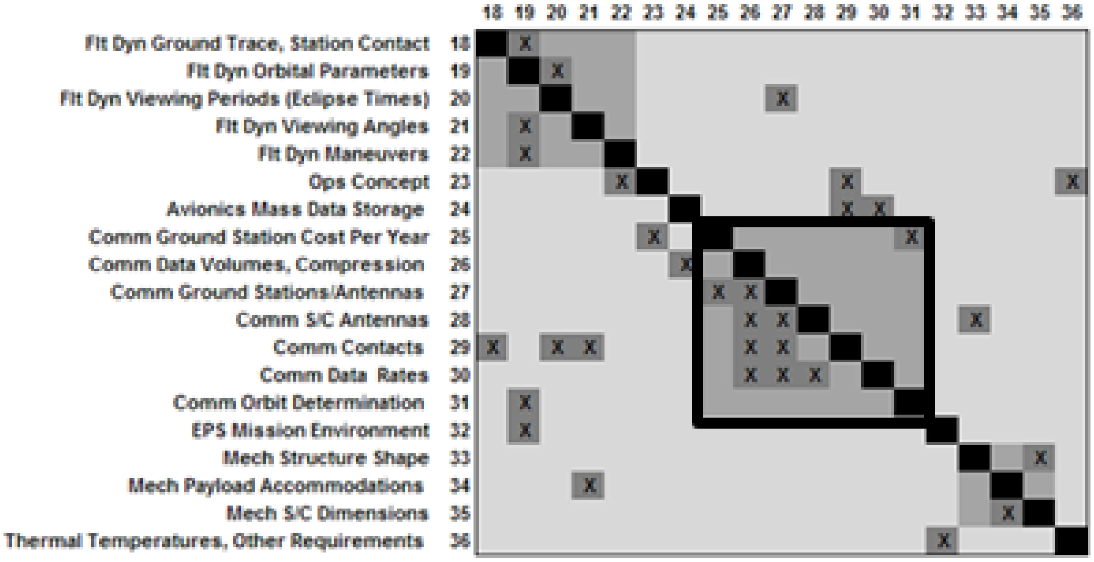
source of such a relationship could be. To provide an explanation for this result, the next subsection returns to an aspect of the design process analysis discussed in Chapter 4.

6.3.2. Communications in the Core of Interdependent Disciplines

Recall from the analysis of the ICE design process discussed in Chapter 4 that a mostly sequential procedure for each design iteration can be defined if certain starting assumptions are made about each discipline's design budgets. After these design budgets were torn from the DSM, the re-partitioned DSM still contained one small coupled block of parameters. Although this core of interdependent disciplines contains some parameters from Avionics, Mechanical, Electrical Power, and Thermal, the critical work that occurs within that block consists of design trades involving all of the engineering design parameters for Flight Dynamics, Mission Operations, and Communications. This result is supported by the actual MDL design process. Discussions with the design team verify that the Communications and Flight Dynamics engineers generally work out important design trades at the beginning of each session. The dependencies in the interdependent core further show that Mission Operations plays a role in those trades and that the resolution of the trades has a direct effect on the design of the other subsystems listed above.

If the sequential information flow observed in the torn and re-partitioned DSM is to be implemented, the trades in this block need to be resolved before the rest of the work begins. According to the particular dependencies within this core, Communications is the central discipline involved in these early design trades. Not only is Communications the only spacecraft subsystem whose design is entirely dependent on the outcome of these trades, but it is also involved in the majority of the total dependencies in this block. Of the 32 marks in the core, 21 represent inputs from, outputs to, or internal dependencies of Communications. Of the 16 interdisciplinary (i.e., off-diagonal) dependencies, 11 are inputs to or outputs from Communications. The diagonal block representing the position of Communications within the core is shown outlined in Figure 6-4.

The central position for Communications in the ICE design process provides a plausible explanation for the indicator role that this subsystem appears to play in the model of shared knowledge. Given that the trades involving the Communications subsystem design need to be resolved early in the process, it is not surprising that the team must have a particularly strong



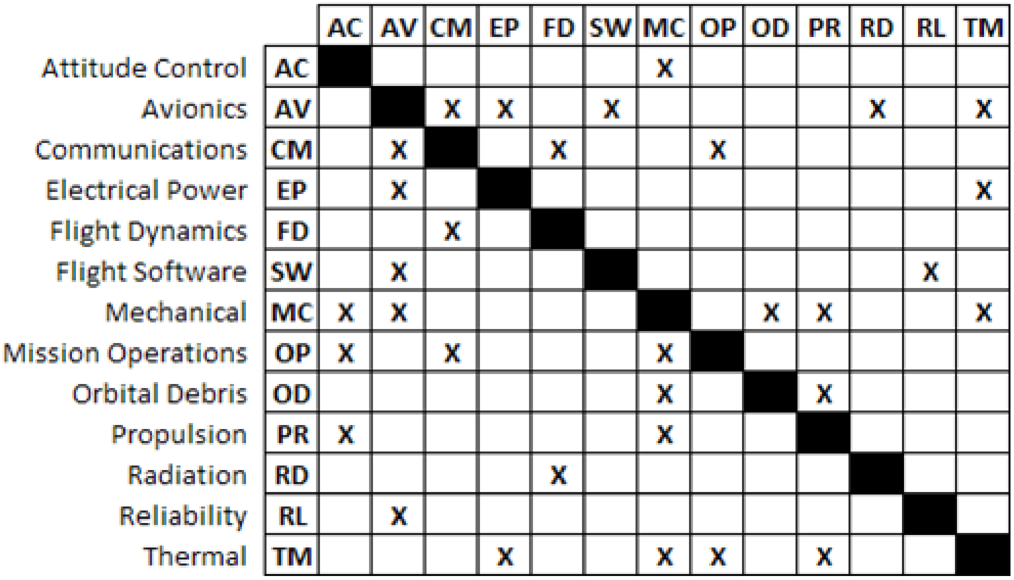
		18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Flt Dyn Ground Trace, Station Contact	18	■	X																	
Flt Dyn Orbital Parameters	19		■	X																
Flt Dyn Viewing Periods (Eclipse Times)	20			■							X									
Flt Dyn Viewing Angles	21		X		■															
Flt Dyn Maneuvers	22		X			■														
Ops Concept	23					X	■						X							X
Avionics Mass Data Storage	24							■					X	X						
Comm Ground Station Cost Per Year	25						X		■						X					
Comm Data Volumes, Compression	26							X		■										
Comm Ground Stations/Antennas	27								X	X	■									
Comm S/C Antennas	28									X	X	■					X			
Comm Contacts	29	X		X	X					X	X		■							
Comm Data Rates	30									X	X	X		■						
Comm Orbit Determination	31		X												■					
EPS Mission Environment	32		X													■				
Mech Structure Shape	33																■		X	
Mech Payload Accommodations	34				X													■		
Mech S/C Dimensions	35																	X	■	
Thermal Temperatures, Other Requirements	36															X				■

Figure 6-4. Communications in the Core of Interdependent Disciplines. Communications is the only spacecraft subsystem whose entire design occurs in the context of the design trades resolved in this core of interdependent disciplines. The outlined block represents the design work for the Communications subsystem within the core. Half of the interdisciplinary information flow in the Core involves inputs to and outputs from Communications.

understanding of the conditions under which this discipline is a major design driver. Thus, the role of Communications in the design process provides some explanation for the connection between this discipline and the dynamics of shared knowledge, as shown in Figure 6-2 and Figure 6-3. Since much of the design work for Communications must be completed early, the team understands the difficult issues associated with its design. Of course, because the DSM was constructed specifically for the typical design process in the Mission Design Laboratory, this core of interdependent disciplines is known to apply only to typical MDL sessions. In the next section, however, an analysis of team coordination applied across all observed design sessions reveals that the DSM could be at least as useful for atypical sessions as it is for typical ones.

6.4. Team Coordination in the Design Process

In the previous section, it was demonstrated that the Communications subsystem plays a central role in the design both from a technical design standpoint and in terms of shared knowledge in the team. This result is notable because it verifies the important role for Communications and shows that a connection exists between shared knowledge and design



		AC	AV	CM	EP	FD	SW	MC	OP	OD	PR	RD	RL	TM
Attitude Control	AC	■						X						
Avionics	AV		■	X	X		X					X		X
Communications	CM		X	■		X			X					
Electrical Power	EP		X		■									X
Flight Dynamics	FD			X		■								
Flight Software	SW		X				■						X	
Mechanical	MC	X	X					■		X	X			X
Mission Operations	OP	X		X				X	■					
Orbital Debris	OD							X		■	X			
Propulsion	PR	X						X			■			
Radiation	RD					X						■		
Reliability	RL		X										■	
Thermal	TM				X			X	X		X			■

Figure 6-5. Expected Interaction Matrix. This is the same matrix as the team-based DSM presented in Chapter 4 but reformatted to be used as a mapping of expected interactions. The primary differences are that it is in alphabetical order by discipline rather than being clustered and that the three disciplines that are not involved in any of the critical design trades (Launch Vehicles, Integration and Test, and Parametric Cost) are not included in the matrix in this form.

process. In this section, the goal is to make the connection between design process and shared knowledge more direct and explicit. Drawing from the literature on the relationship between product architecture and organizational structure, the analysis connects the team-based DSM presented in Chapter 4 to reported interactions among the members of the team by computing a metric of socio-technical congruence (STC). This metric is then related to the dynamics of shared knowledge, which demonstrates that the architecture-organization connection is associated with shared knowledge in the team.

6.4.1. Team Dynamics and the Design Process

Recall the team-based DSM presented in Chapter 4. Because this representation of interdependent disciplines was constructed from critical design trades, it represents the interactions that are expected to take place during a typical design session. Figure 6-5 shows this same matrix again but this time reformatted as an *expected interaction matrix*. The primary differences are that it is in alphabetical order by discipline rather than being clustered and that the three disciplines that are not involved in any of the critical design trades (Launch Vehicles, Integration and Test, and Parametric Cost) are not included in the matrix in this form.

The purpose of restructuring the matrix of technical interactions in this way is to compare it to reported interactions among the members of the design team. To conduct this comparative analysis, a matrix of reported interactions was constructed for each design session. Because the ICE environment is characterized by constant interactions and ubiquitous information flow, it was not feasible to construct a network of actual interactions in the design team. Without an intensive analysis involving audio recordings of each design session, any attempt to catalog all interactions among the team would be futile. For this reason, data on team interactions were collected using a question on the post-session survey.[22] This question asked each member of the team to rate the importance of interactions with each other member of the team. They were given four choices: 3 – Essential, 2 – Important, 1 – Helpful, and 0 – Unnecessary (blank responses were taken to be equivalent to Unnecessary).[23] Based on these data, a team-based DSM of reported interpersonal interactions was constructed.

This DSM of reported interactions is the adjacency matrix of a weighted network with edge weights ranging from 1 to 3. The expected interaction matrix, on the other hand, is unweighted since it simply indicates the direction of information flow within the critical design trades. The two data sets cannot be compared without an adjustment being made to one of them. One possible solution to this problem is to somehow quantify the importance of flows in the loops representing the trades. The difficulty with this approach is that there is no consistent and objective way to weight these flows. In fact, this is why the full parameter-based DSM is binary. Any weighting of the loops would require a judgment of the importance of each parameter and the importance of the flows between parameters. Since the parameters are not all defined at the same level of granularity, such distinctions would not be meaningful.

Therefore, the best approach to comparing the two matrices is to dichotomize the data for the DSM of reported interactions. Although this results in the loss of some information, the distinctions are based on individuals' subjective views about the relative meanings of the words "Essential", "Important", and "Helpful" (presumably, the meaning of "Unnecessary" is less

---

[22] The question on team interactions was added to the survey beginning with session 2 because it was deemed to be important following the observation of session 1. This highlights one of the particular benefits of supplementing the interview, survey, and documentary data with observations of the full design sessions.

[23] To test for randomness in the survey responses, a chi-square (χ^2) goodness-of-fit test was conducted for each of the 11 design sessions for which interaction data were collected. Based on a random distribution in which the four options are equally likely, the χ^2 statistic is significant to a level of $p \leq 0.01$ in all 11 cases ($p < 0.01$ for 10 cases and $p = 0.01$ for one). Thus, it is concluded that the survey responses were not given at random.

ambiguous). Clearly, any rating of Essential or Important can be assumed to indicate that the interaction should occur, and a rating of Helpful implies that the interaction is not required for completion of the work. To make the matrix of reported interactions directly comparable to the matrix of expected interactions, responses of Essential and Important are marked (i.e., Needed), and responses of Helpful and Unnecessary are not marked (i.e., Not Needed).

Finally, one last adjustment was made to the two matrices because some design sessions did not require participation of one or more disciplines. For example, the lunar surface concept generation session required neither Attitude Control (which is not relevant once the hardware reaches the surface) nor Parametric Cost (since there was no point design on which to base an estimate). When a discipline was not involved, no parameter flow data were collected on that discipline or on others' interactions with that discipline. The matrices of reported and expected interactions were reconstructed given the absence of that discipline. This meant both removing the row and column corresponding to that discipline and removing any expected interactions among other disciplines in loops that no longer existed after that discipline was removed.[24]

Based on the above procedure, a *reported interaction matrix* was constructed for each design session and compared to the expected interaction matrix. The initial comparison was done by overlaying the two matrices according to a framework used by Sosa et al. (2003). The result, which is referred to here as the *congruence matrix*, can be used to highlight four distinct cases in mapping expected to reported communication: # (design dependency exists, and the corresponding interaction takes place), X (design dependency exists, but no interaction takes place), O (no design dependency exists, but an interaction takes place), and <blank> (no design dependency exists, and no interaction takes place). A congruence matrix comparing reported interactions for one MDL design session (session 3 among those observed) to expected interactions is shown in Figure 6-6.

Along with the congruence matrix shown in Figure 6-6(a), a summary of the statistics for the counts of the cell values is given in Figure 6-6(b). The summary is shown here as a 2 *x* 2 contingency table, which can be used as the basis for a chi-square (χ^2) test to determine whether the matrix of reported interactions is related to the matrix of technically expected interactions.

---

[24] In the rare case in which interaction data was unavailable for a discipline involved in the session, the discipline had to be removed to maintain consistent matching between the two matrices. Other disciplines' interactions were not removed in these cases because the loops involving the missing disciplines were still expected to take place during the design session.

		AC	AV	CM	EP	FD	SW	MC	OP	OD	PR	RD	RL	TM
Attitude Control	AC	■	O		O	O	O	#			O			
Avionics	AV		■	#	#		#				O	#		#
Communications	CM		#	■	O	#		O	#			O		
Electrical Power	EP	O	#	O	■	O	O	O	O					#
Flight Dynamics	FD	O		X	O	■					O			
Flight Software	SW	O	#	O			■						X	O
Mechanical	MC	#	#	O	O			■		X	#			#
Mission Operations	OP	X	O	#		O	O	X	■					
Orbital Debris	OD	O			O			#		■	#			
Propulsion	PR	#				O		#		O	■			
Radiation	RD		O			X						■		
Reliability	RL	O	#	O	O	O					O		■	O
Thermal	TM	O	O	O	#	O	O	#	#		#		O	■

(a)

Actual Interactions	NO (87)	X (6)	(81)
	YES (69)	# (26)	O (43)
		YES (32)	NO (124)
		Expected Interactions	

χ^2	*p*
22.4	< 0.001

(b)

Figure 6-6. Expected and Reported Team Interactions for an MDL Design Session. The framework is modeled after the structure proposed by Sosa et al. (2003). **(a) Congruence Matrix.** The matrix shows an overlay of the expected interaction matrix and the reported interaction matrix for session 3. X indicates an expected interaction but no reported interaction, O a reported interaction but no expected interaction, and # both a reported and an expected interaction. A blank cell indicates that there was no interaction either expected or reported. **(b) Congruence Matrix Statistics.** The 2 *x* 2 contingency table on the left indicates the number of each type of mark in the matrix, and the table on the right shows the χ^2 statistic and its significance.

For this test, the null hypothesis states that the reported interactions are not related to the DSM-based expected interactions. Running the test on each of the 11 design sessions, it is found that the χ^2 statistic is significant for nine of the 11 cases. For eight of those sessions, it is determined that $p \leq 0.01$. In those cases, the null hypothesis is rejected, and it is concluded that reported interactions are related to expected interactions. In one case (session 5), the χ^2 statistic is significant to a level of $p = 0.05$. With Yates' correction for continuity, however, the statistic is no longer significant for that session. The two cases for which the statistic is not significant both with and without the correction are sessions 2 and 11. For sessions 2, 5, and 11, no peculiar characteristics could be identified to explain the different pattern of reported interactions as compared to expected interactions for each of these sessions individually. To determine the reasons for the different relationships between reported and expected interactions across design sessions, further work would need to account for more subtle differences among the sessions than those used in this research. For the present purposes, the effect of these results on the analysis of team interactions across all design sessions will be discussed in section 6.4.3.

6.4.2. Definition of Socio-Technical Congruence

In a study of communications in a distributed software development team, Cataldo et al. (2008) developed a metric to quantify the relationship between expected and reported team interactions. This metric, which the authors termed *socio-technical congruence*, is defined as the ratio of the number of expected interactions that actually occur to the total number of expected interactions. Based on the formalism of Figure 6-6, the metric is computed by dividing the number of #s in the congruence matrix by the sum of the number of #s and the number of Xs.

In the analysis of the ICE environment, a metric of socio-technical congruence is computed for each observed design session. The metric used in this thesis, however, is not exactly the same as the one used by Cataldo et al. (2008). Because they analyzed a team in a distributed environment, all team interactions are implicitly considered to be deliberate and purposeful. In the ICE environment, on the other hand, the design team is collocated for virtually the entire project. For this reason, interactions can occur without premeditation or specific purpose. This difference introduces a new variable to the calculation – the number of interactions that occur unnecessarily.

This fundamental difference between these two types of design environments is related to the structure of Integrated Concurrent Engineering. Recall from Chapter 3 that ICE is effectively a lean engineering environment. For this reason, the metric for socio-technical congruence as applied to ICE is computed according to lean principles. The metric proposed by Cataldo et al. (2008) considers only expected interactions that actually occur. It does not include those interactions that are unnecessary but occur anyway. According to lean thinking, these interactions are classified as waste and ought to be included in the calculation of socio-technical congruence. With this modification, the metric used in this thesis includes the sum of # cells and blank cells in the congruence matrix. This sum is then divided by the number of all possible interactions, i.e., the total number of non-diagonal cells in the matrix. Formally, socio-technical congruence is defined here as

$$C_{S-T} = \frac{N_{\#} + N_{b}}{N}, \tag{6-2}$$

where $N_{\#}$ is the number of cells marked #, N_b is the number of non-diagonal blank cells, and N is the total number of non-diagonal cells in the matrix. This metric is a simple cell-by-cell matching of the expected interaction matrix to the reported interaction matrix.

In examining the structure of Eq. (6-2), the reader might question the equal weighting of the # cells and the blank cells. Even though lean principles would suggest that the number of unexpected interactions that do not occur (N_b) should be a factor in the calculation, it is arguable whether they should have the same importance as expected interactions that do occur ($N_{\#}$). Thus, it is possible that the calculation of $C_{S\text{-}T}$ could be made more robust by placing a coefficient, probably with a value between 0 and 1, in front of N_b. Although a dedicated study on socio-technical congruence in lean engineering environments might yield the appropriate value of such a coefficient in various contexts, any such coefficient chosen for the present purposes would be arbitrary. For this reason, the calculation used here is done without a coefficient (or, effectively, with a coefficient of one), giving equal weighting to unexpected and expected interactions.

Alternatively, it might also be argued that the quadratic assignment procedure (QAP), as discussed in Chapter 5, could be used to measure structural similarity between the reported and expected interaction matrices. In that case, the QAP correlation would be used in place of the measure of socio-technical congruence described above. That metric is chosen instead of the QAP correlation because it is intended as a modification of the metric proposed by Cataldo et al. (2008), which was constructed as a ratio of expected and actual interactions. The QAP correlation was used as the measure of stability of shared knowledge, on the other hand, because that metric is explicitly meant to quantify how the structure of the network changes.

Despite this difference in the nature of these measured phenomena, a comparison of the two metrics of team coordination was made to test the model. According to this comparison, the values for socio-technical congruence measured as in Eq. (6-2) and by using the QAP correlation are closely related to each other. Over the course of the 11 design sessions for which team interaction data were available, the correlation between the two alternative metrics is $r = 0.798$ and $p < 0.01$. To complement this test, the next subsection also includes a brief assessment of how the relationship between socio-technical congruence and shared knowledge changes if the QAP correlation is used as the metric.

Regardless of how it is measured, however, socio-technical congruence in itself is meaningful only to the extent that it affects or is affected by the way that the design team members *think* about their work. After all, it is the thoughts of the engineers that ultimately become formalized and codified into an actual design. The next subsection explores this assertion by examining the relationship between socio-technical congruence and the dynamics of shared knowledge in the team.

6.4.3. Socio-Technical Congruence and the Dynamics of Shared Knowledge

The purpose of this subsection is to demonstrate a relationship between socio-technical congruence and the dynamics of shared knowledge. Before this can be done, it is important to first show that the values of $C_{S\text{-}T}$ do not result from the team members' experience with the survey (just as was done for ΔS in Chapter 5). Calculating Spearman's ρ for the relationship between $C_{S\text{-}T}$ and design session sequence, it is found that $\rho = 0.218$ and $p = 0.52$. Given this result, the value of socio-technical congruence is determined to be independent of the respondents' experience with the survey question and thus is directly related to the content of the design sessions.

Based on the 11 ICE design sessions for which the necessary data were collected, Figure 6-7 shows the nature of the relationship between $C_{S\text{-}T}$ and ΔS. In the figure, socio-technical congruence is labeled as team coordination, a term that implies not only communication among the team members but also their efforts to coordinate around the documented design process. Overall, the figure demonstrates a statistically significant relationship across the design sessions studied. The trend indicates that shared knowledge in the team changes most when the team engages primarily in those interactions that are expected to occur in the technical design. Because of the nature of the data collected, a finer distinction among the data points has also been made. Recall that the DSM was constructed to depict technical information flow in a typical MDL session. It would be reasonable to assume that the expected interaction matrix applies only to that type of session. As the figure shows, though, a statistically significant trend

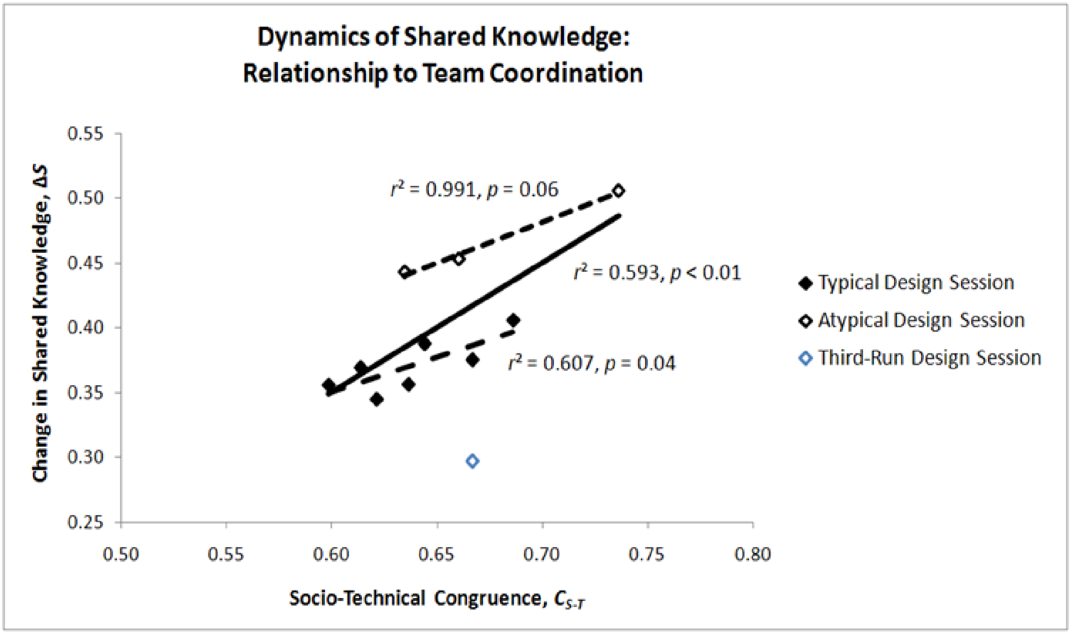


Figure 6-7. Dynamics of Shared Knowledge and Design Team Coordination. Overall, the trend indicates that shared knowledge in the team converges most when the team engages primarily in expected interactions. Interestingly, the trend is most significant when no distinction is made between typical and atypical sessions even though the expected interactions are based entirely on typical design sessions.

also exists when no distinction is made between typical and atypical sessions. This suggests that the documented interactions for typical sessions generally apply to atypical sessions as well.

The results reached by examining the two types of sessions separately should also be considered. Among only typical design sessions, a somewhat significant correlation exists between $C_{S\text{-}T}$ and ΔS, though the p-value of 0.04 is relatively close to the 0.05 cutoff for statistical significance. For the advanced sessions, the trend among only three data points is nearly statistically significant with $p = 0.06$. This is notable, though, because p-values are highly sensitive to changes in a single data point when so few are available. If just one additional advanced session were included or if just one of the three points were shifted slightly, the result could become highly statistically significant (of course, it could also become unambiguously insignificant). Thus, the position of the three points does not provide conclusive evidence of a trend for advanced sessions, but it does suggest at least some possibility that such a relationship exists. If this apparent trend is indeed present (which would require several more data points to demonstrate), it would indicate that a similar relationship exists for typical and advanced sessions separately but that it is shifted upward along the ΔS axis for atypical sessions.

Of course, this analysis was conducted using all 10 first-run design sessions. Recall from section 6.4.1, however, that reported interactions are not shown by the χ^2 test to be related to expected interactions for two of the design sessions (or three when accounting for Yates' correction). If it is assumed that reported interactions must be related to expected interactions for socio-technical congruence to be a meaningful construct, then the analysis must be repeated with the points that do not meet that criterion removed from the data set. First, sessions 2 and 11 were removed since reported interactions for those sessions were not shown to be related to expected interactions both with and without Yates' correction. After removing these points, a correlation can no longer be computed for atypical sessions alone because only two such data points remain. For typical sessions alone, the r-squared for the regression is $r^2 = 0.656$ and $p = 0.05$. For all 8 remaining sessions, $r^2 = 0.734$ and $p < 0.01$. Thus, the results do not change significantly when those two points are removed. Next, the data point for session 5 was also removed because reported interactions for that session were not shown to be related to expected interactions when Yates' correction was made. After removing this third point, a statistically significant trend no longer exists for the typical sessions alone. Still, the overall trend for the remaining first-run sessions still exists, and the regression yields $r^2 = 0.717$ and $p = 0.02$. This result demonstrates that the relationship between $C_{S\text{-}T}$ and ΔS still exists even when the correlation includes only those sessions for which reported and expected interactions are related to each other according to the χ^2 test.

To evaluate the robustness of the results in another way, the relationships were also computed when using the QAP correlation in place of socio-technical congruence. The results of this analysis are shown in Figure 6-8. The general trend for the relationship between team coordination and change in shared knowledge remains the same for typical design sessions.[25] The nearly statistically significant trend for atypical sessions measured with only three available data points becomes more tenuous but is still at least somewhat plausible. The overall relationship that includes both typical and atypical design sessions, however, no longer exists in this case. This difference could be a mere artifact of the calculation, or it could imply some fundamental difference in team coordination when viewed as a ratio of expected and reported

---

[25] If both the QAP correlation is used and the data points for sessions 2, 5, and 11 are removed, a statistically significant relationship no longer exists. This occurs, however, only if both of these conditions are applied to the model. With only one of these variations applied, the results do not materially change.

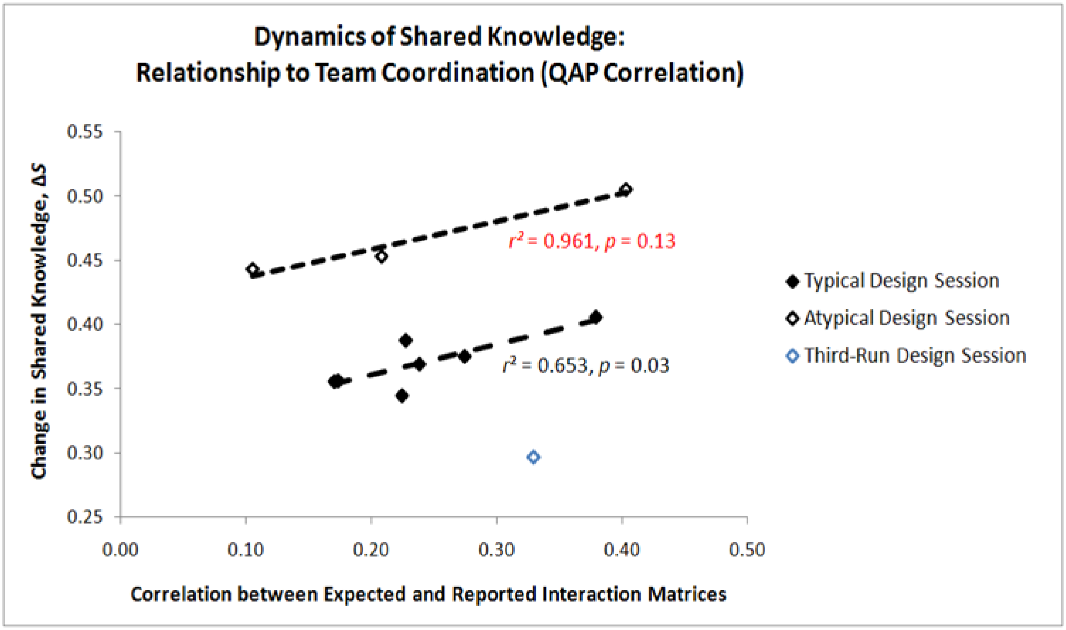


Figure 6-8. Measuring Team Coordination as a QAP Correlation. The relationship between team coordination and change in shared knowledge is independent of the method for measuring team coordination when measuring for typical design sessions but not for atypical sessions. The nearly statistically significant trend for atypical sessions given the three data points available becomes more tenuous in this case, but it still appears at least somewhat plausible that such a trend might exist given more data on atypical sessions. The r^2 and p-value are shown in red font to indicate this. The relationship between team coordination and change in shared knowledge across both typical and atypical sessions, however, is entirely absent when using the QAP correlation to measure team coordination.

interactions (the proposed metric) and when viewed as a structural property of the expected and reported team interaction networks (the alternative QAP metric).

Recall the discussion of section 6.1, which was largely based on the previously demonstrated result that a change in shared knowledge constitutes an increase in shared knowledge, or learning. According to this and the observed relationship between $C_{S\text{-}T}$ and ΔS, shared knowledge increases most when socio-technical congruence is high, i.e., when reported interactions map closely to expected interactions. Although these results are merely correlations and thus do not imply causality, they do indicate that the team members learn more and thus could be more productive when they interact as they are "supposed to" according to the DSM representation of the technical design process. Conversely, they might interact as expected because they are learning, but the association between $C_{S\text{-}T}$ and ΔS exists either way. Surprisingly, this result is applicable to all of the observed design sessions despite the fact that the expected interaction matrix was constructed from information flow in a typical session.

Since this trend does not appear when using the QAP correlation to measure socio-technical congruence, however, the existence of a relationship between team coordination and change in shared knowledge across all session types is somewhat questionable. Still, the trend using the measure of socio-technical congruence proposed in Eq. (6-2) suggests that the DSM is generally applicable to both typical and atypical MDL design sessions even though it was initially constructed to represent only typical sessions. Moreover, in combination with the observation that a greater amount of team learning occurs during atypical design sessions than during typical ones (see Figure 6-1), the results presented in this section lead to the counter-intuitive conclusion that following the patterns dictated by the DSM representation of typical information flow may actually be most important for the *less typical* design sessions. This result lends some support to the proposition that the DSM constructed for the typical ICE process can be applied to other settings if certain adjustments are made on a case-by-case basis.

6.5. The Role of Shared Knowledge in Engineering Design

Recall the relationships demonstrated in Figure 6-1. According to the graphs shown in that figure, the dynamics of shared knowledge and the product of the design are strongly correlated. Figure 6-7 demonstrates a similar relationship – that the dynamics of shared knowledge and team coordination are also strongly correlated. These two figures together show that a correlation exists between shared knowledge and two distinct sets of metrics – one regarding team dynamics and the other related to the technical system. Thus, to fully describe the nature of the relationships among these three properties of the design process, it is necessary to determine whether team coordination and the design product are directly related to each other.

Based on the data presented in Figure 6-9, a statistically significant relationship was not found between team coordination and MTRL using Spearman's ρ. A statistically significant correlation was found between $C_{S\text{-}T}$ and each of the other three system attributes, but there are two features that distinguish these plots from the ones in Figure 6-1 comparing ΔS to each of the technical attributes. First, the p-values, while well below the $p = 0.05$ accepted cutoff for statistical significance, are generally higher than the equivalent p-values in the shared knowledge plots. Second, the correlations in this case are all highly dependent on a single extreme data point – the one corresponding to session 9, the advanced lunar surface concept generation

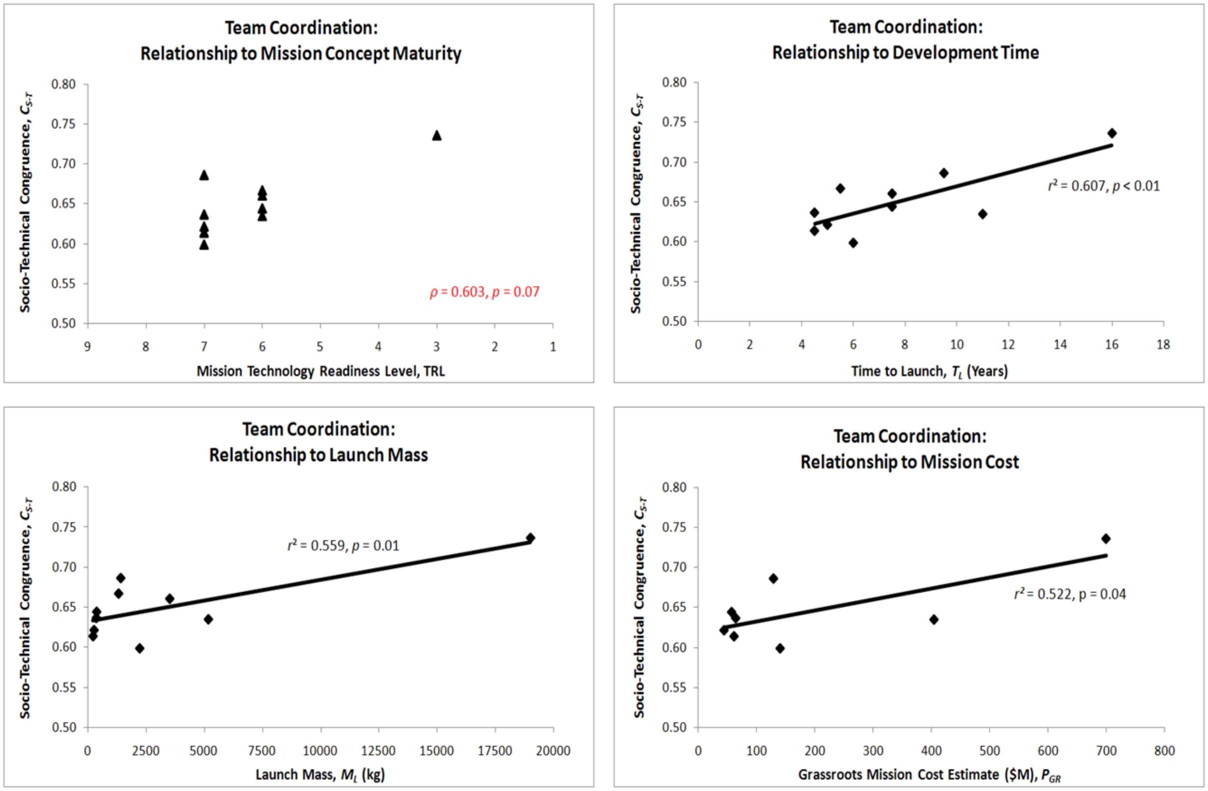


Figure 6-9. Team Coordination and the Product of the Design. A statistically significant relationship was not found between team coordination and MTRL (as indicated by the red font in the figure). Team coordination is correlated with system development time, launch mass, and mission cost, but those correlations are all dependent on a single extreme data point. In the absence of that point, the correlations would no longer exist. Thus, team coordination could be directly related to features of the design product, but the existence of that relationship is not conclusive from the available data.

session. It is immediately obvious by visual inspection that the relationships of team coordination with system development time, launch mass, and mission cost no longer exist if that extreme point is removed from the data set. This is not the case in the plots in Figure 6-1. For all three metrics, the relationship with ΔS shown in that figure is not dependent on the extreme data point.

Figure 6-10 conceptually summarizes the findings described above. As the figure shows, shared knowledge is related to both team coordination and the design product, but the relationship between those other two metrics is less convincing. This phenomenon could imply a number of possible ways in which these three general properties of the process affect each other. For instance, team coordination and the design product might independently affect (or be affected by) the shaping of shared knowledge in the team.

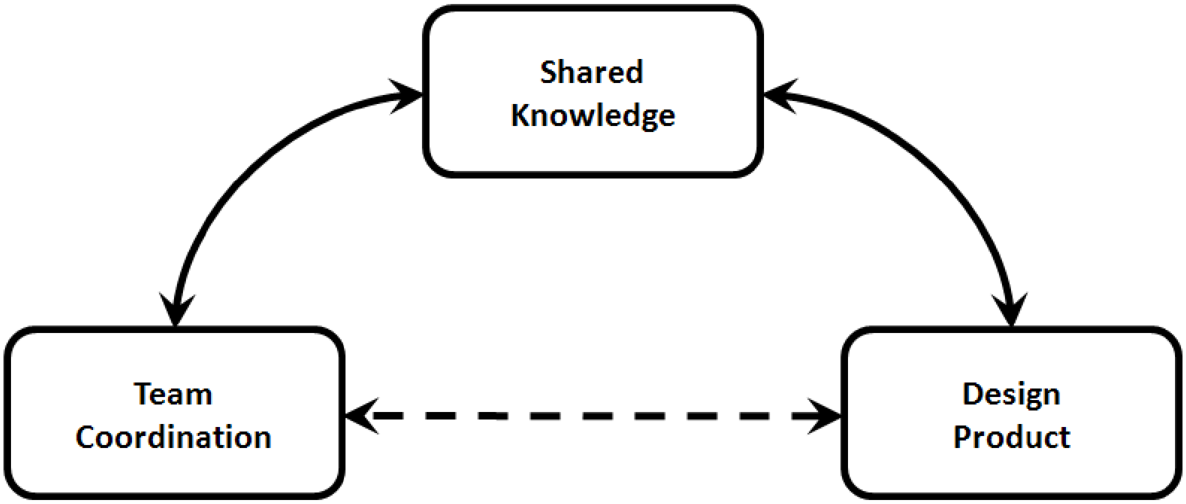


Figure 6-10. The Role of Shared Knowledge in Engineering Design. The solid lines depict the strong relationships of shared knowledge with team coordination (i.e., socio-technical congruence) and with system-level technical attributes of the design product (MTRL, system development time, launch mass, and mission cost). The dashed line indicates that a direct relationship between team coordination and the design product might exist but that the relationship is not conclusive from the available data.

Alternatively, it is possible that one of the three properties could be either a mediator or a moderator of the relationship between the other two. Baron and Kenny (1986) offer a complete discussion of the distinctions between mediating and moderating variables. The statistical tests for moderation and mediation cannot be readily applied to the data presented in this research because the distinctions between independent and dependent variables have not been established, i.e., the results show only correlations and not causations. Depending on the direction of causality, the relationships shown in Figure 6-10 could indicate that a moderating or mediating relationship exists. Thus, shared knowledge may be a mechanism by which team coordination is translated into the product of the design (mediating relationship), or the extent of the influence of team coordination on the design product could be dependent on the how shared knowledge changes over time (moderating relationship). In addition, either of these relationships could occur in the opposite direction. A study on the precise nature of these interrelationships and the direction of causality among shared knowledge, team coordination, and the design product is an important area of future work following from this thesis.

6.6. Review of Information Flow and Shared Knowledge in Design Teams

This chapter has offered a set of interdisciplinary insights about the connection between the space mission design process and shared knowledge in the design team. The discussion started with the relationship between the dynamics of shared knowledge and the product of the design, and it ended with the relationship of each of those variables to team coordination. Between those discussions, the chapter explored the types of knowledge in the content of shared mental models in space mission design and the special role for one subsystem – Communications – in the design process. Taken together, the analyses presented in this chapter support the argument of Klimoski and Mohammed (1994) that shared knowledge is not merely a metaphor or an abstract notion of little operational value but rather a real property of the team that has implications both for their work and for the product of that work. More importantly for the design of engineered systems, the conclusions of this chapter demonstrate perhaps the first complete and systematic study that empirically demonstrates the strength of the argument of Badke-Schaub et al. (2007) that shared mental models can be applied meaningfully to engineering design.

This chapter represents the last of the formal analyses presented in this thesis. The final part of the thesis, consisting of two chapters, integrates all of the analyses at a high level and provides conclusions, implications, and future work. The purpose of Chapter 7 is to demonstrate the direct and practical relevance of the research to the ICE environment and to the Mission Design Laboratory in particular. The chapter first discusses implications and recommendations regarding each of the four elements of the MDL: People Process, Tools, and Facility. Then, it offers an integrated model depicting the standard ICE design process using the insights and recommendations that come from this research. After that, Chapter 8 concludes the thesis with a broader look at the implications of the research at all levels, including ICE, space systems design in general, and the design and development of other complex engineered systems. That chapter also includes the contributions of this research to several academic fields from which it has drawn and to the emerging field of Engineering Systems. Finally, the many possible directions for future work in each of those fields are explored.

[This page intentionally left blank]

Chapter 7

Implications for the Future of Integrated Concurrent Engineering

In the last three chapters, two separate models of the Integrated Concurrent Engineering (ICE) design environment – one on process and one on people – were presented and then unified. From that analysis, the thesis has established a set of findings about the nature of technical information flow in the design process, the dynamics of shared knowledge in the team, and the relationship between the two. The goal of this chapter is to codify these interdisciplinary insights into a set of guidelines and recommendations for improving the ICE design process in the future. Section 7.1 provides a concise overview and summary of the results of the thesis. Then, section 7.2 offers concrete recommendations for the Mission Design Laboratory (MDL) structured around the four elements of that design center: People, Process, Tools, and Facility. Section 7.3 synthesizes the recommendations into a standardized design process model for ICE. Following that, section 7.4 briefly discusses the implications for establishing new ICE design centers and offers some suggestions for approaching such a project. Finally, section 7.5 closes the chapter with a few thoughts about the role of shared knowledge and cognition in full-scale space systems development programs and other large organizations.

7.1. Shared Knowledge and the ICE Design Process: Overview of the Results

The purpose of this section is to provide a simple and accessible summary of the results presented in the three previous chapters. Chapter 4 offered a set of guiding principles for constructing a Design Structure Matrix (DSM) for the ICE environment. Based on the DSM analysis, the thesis identified the phases of the ICE design life cycle, the critical design trades and interdependent disciplines, a set of starting assumptions that can be made at the outset of the design, and the process of sequentially executed design iterations that results from making those assumptions. In addition, the analysis has shown that the Communications subsystem plays a central role in the core of interdependent disciplines in the DSM.

Chapter 5 developed a new way of analyzing shared knowledge in teams – the structural approach. Structural analysis of the dynamics of shared knowledge revealed a connection between change in shared knowledge in the team and several technical features of the design process, including mission concept maturity, system development time, launch mass, and mission cost. In addition, it was shown that a change in shared knowledge across the team generally corresponds to an overall convergence of shared knowledge. Then, in Chapter 6, these technical features were examined together. The comparison among those metrics, combined with the results discussed above, suggested that change in shared knowledge increases with the complexity of the system, where complexity is defined as a catch-all property that maps to those other technical attributes.

After the relationships between shared knowledge and the system-level attributes were identified, other relationships were found regarding the content of shared mental models. This analysis revealed that the Communications subsystem plays an important role as part of the content of shared knowledge. Both the Technology Readiness Level (TRL) of Communications and the team's overall assessment of its importance were shown to be strongly correlated with shared knowledge in the team. This implies that the Communications subsystem is, in some way, an indicator of shared knowledge. As a result, it is particularly important that the team as a whole understand the design of the Communications subsystem. This interpretation is consistent with the central role of that subsystem in the interdependent core found in the technical analysis in Chapter 4.

The last part of the analysis demonstrated a relationship between the dynamics of shared knowledge and team coordination. Specifically, it was shown that change in shared knowledge and a measure of team coordination called socio-technical congruence (the extent to which reported interactions match those expected in the technical design process) are positively correlated. Given the result that change in shared knowledge is equivalent to a convergence of knowledge (or team learning), it can be inferred that the team members learn the most from their work when they interact in the manner prescribed by the mapping of the technical design process. Because the results are correlations, however, the causal relation is not necessarily clear. Team learning might result from team coordination, or the team might interact as expected because they are learning. Finally, combining this with previous results, it was shown that shared knowledge is related both to team coordination and to the design product, though the evidence for a direct relationship between team coordination and design product is somewhat less convincing.

These results are based entirely on analysis of data collected in one particular design setting, the Mission Design Laboratory. Following from this analysis is a set of recommendations that can be applied directly to that specific environment. These recommendations cannot be implemented as-is in any other context, but they form a basis for guidelines about the design process more generally. The remainder of this chapter presents the implications of the research in increasing generality. First, a set of seven recommendations that apply exclusively to the Mission Design Laboratory is presented. Then, these recommendations are integrated to provide a more general standardized model of the typical MDL design process. Using this model, the discussion is then broadened to include the implications for the establishment of new ICE design centers. Finally, the results and recommendations are considered in the context of larger systems and engineering organizations.

7.2. Recommendations for the Mission Design Laboratory

This section is intended to serve as a quick reference guide for the MDL management, discipline engineers, and customers. It contains substantive recommendations for the MDL process that are directly supported by the analysis summarized in the previous section. Although these recommendations are not made on the basis of any quantifiable metric of performance or quality in the design sessions, they are based on empirical data on the team and on the process. Thus, the recommendations cannot be taken as hard and fast rules but rather as suggestions for implementation and testing in actual design sessions. The discussion is organized according to the four elements of the MDL: People, Process, Tools, and Facility. Many of these recommendations correspond partially to existing practice in the MDL. The recommendations listed here supplement these practices because they are provided in a formal structure and are based on systematic analysis of the work of the MDL.

7.2.1. People

The results of the shared knowledge portion of this thesis have led to an important conclusion about how the people in the MDL work. Specifically, it was found that the members of the team learn the most (i.e., shared knowledge converges) either from each other or from exposure to the same external information during the design of the most complex systems. For less complex systems, shared knowledge remains relatively static over the course of the work. In these cases, the team members may already know as much as they need to know before the session begins and thus can “hit the ground running.” For complex systems, on the other hand, a certain amount of planning prior to the start of the design work could help to contribute to a more productive session. As discussed in Chapter 6, this type of planning can help the team to improve its overall knowledge of the mission concept and thus can lead to improved outcomes. Even for less complex mission concepts, planning can also be helpful – both by reducing the risk of lock-in on flawed shared mental models (i.e., groupthink) and by creating an opportunity for more innovative thinking about an otherwise routine task.

A pre-work meeting similar to the one described above already takes place before each MDL design session, usually during the week before the study takes place.[26] During this meeting, the customer team clarifies the requirements, and certain members of the design team give some initial consideration to issues that are expected to be particularly important. This meeting helps the Team Lead and the Systems Engineer to establish the general direction of the work and provides an opportunity for some of the disciplines to begin collaborating, but it does not serve the same function as the planning step that follows from the analysis of this thesis.

Assuming that the increase in shared knowledge observed in this research occurs because of the learning that is necessary for execution of the collaborative design effort, the results of the research suggest that design outcomes might be improved through the establishment of a well-defined period of learning and consensus building at the beginning of each design session. The purpose of such a period is neither to plan specific interactions nor to discuss abstract issues. Rather, it is to consider the variety of ways in which the session might progress and to develop a list of potential design hurdles to consider. Most importantly, however, a period of learning and

---

[26] A post-work meeting is also conducted after each design session. This is important for integration of the design session report and certain administrative tasks, but it does not follow directly from the analysis presented in the thesis.

consensus building would give every team member the opportunity to study the entire mission concept and to assess the potential interdisciplinary design issues that they will face before the fast-paced, deliverables-driven design work begins.

To understand how the period of learning and consensus building can be implemented, consider how an MDL design session normally begins. On the first day of the session (usually Monday morning), the customer team gives a presentation to the design team in which they enumerate and explain their requirements, expectations, and aspirations for the design work. Over the course of the week, the design team (with the customers' active involvement) carries out several iterations of the design. If a team member's knowledge and perceptions change as a result of more completely understanding the customer's expectations (or if those expectations change), it would be helpful for the rest of the team to understand how that person's perceptions are affected. This is particularly true if the new knowledge affects the perceptions of several team members. The period of learning and consensus building can help to ensure that newly formed views are discussed and placed in proper perspective before the full design work begins.

The period of learning and consensus building would need to be held in addition to the pre-work meetings that already occur in the MDL. The aim of those meetings is generally to exchange information between the customer team and a subset of the design team. In contrast, the period of learning and consensus building would necessarily include the entire team. Thus, the first recommendation to the MDL is

R1: ***Schedule a period of learning and consensus building prior to the start of each design session. The importance, depth, and duration of this period will vary with the expected difficulty or complexity of the mission concept under study.***

Once the design work has begun, the team can be organized in a way that best leverages the interdependence among certain disciplines. The clustered team-based DSM, shown in network form in Figure 7-1, offers a possible means of organizing the design team into sub-teams. Although these sub-teams would facilitate critical interactions among the disciplines that are most tightly interconnected in the technical design, they are not meant to exclude interactions across sub-team lines. On the contrary, the interactions that occur between the groupings could

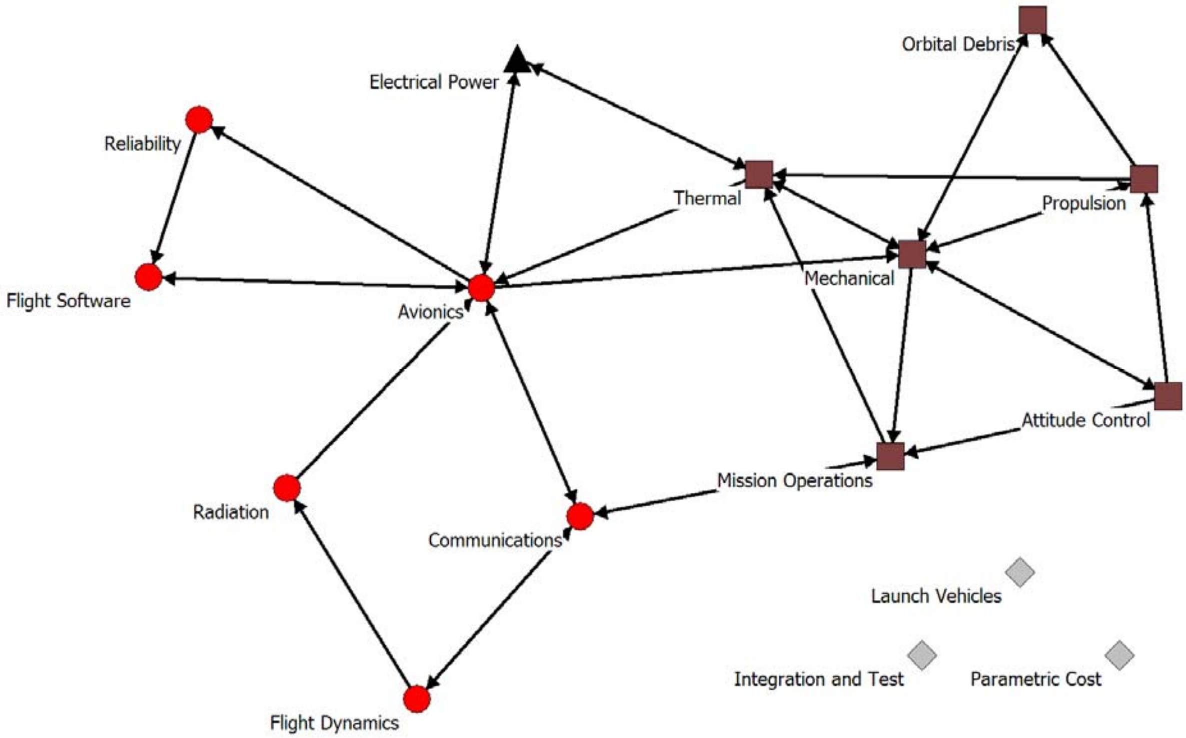


Figure 7-1. Sub-Teams in the Mission Design Laboratory. This network graph shows the same arrangement of team members as the clustered team-based DSM presented in Chapter 4. The sub-teams are indicated by color and shape: Sub-Team 1 as red circles, Sub-Team 2 as brown squares, Sub-Team 3 as a black triangle, and disconnected disciplines as gray diamonds.

serve as critical interfaces between the sub-teams. Even beyond this, however, it will still be necessary for each team member to collect and distribute the required pieces of information that are not represented in the structure of the team-based DSM. To facilitate this, the full parameter-based DSM would serve as a useful tool to complement the sub-team structure.

The clustered team-based DSM reveals three sub-teams in the typical MDL process. Sub-Team 1 is generally composed of disciplines whose chief concern is mission environment, i.e., the spacecraft's location. The disciplines involved in this sub-team are

- Flight Dynamics
- Communications
- Radiation
- Avionics
- Flight Software
- Reliability

Sub-Team 2, on the other hand, deals more with the physical hardware of the spacecraft. The disciplines involved in this sub-team are

- Mission Operations
- Attitude Control
- Mechanical
- Propulsion
- Thermal
- Orbital Debris

Logically, Mission Operations could be placed in either sub-team, but the clustering procedure placed it in Sub-Team 2 because it is connected to more disciplines in that cluster.

Sub-Team 3 consists of just one discipline: Electrical Power. This is a reasonable assignment because that discipline is dependent on both orbital dynamics issues (exposure to sunlight) and hardware issues (solar array size). Thus, this engineer would not be expected to work in isolation. On the contrary, he or she would work closely with both of the other sub-teams. In addition, recall that because Launch Vehicles, Integration and Test, and Parametric Cost are not involved in any loops, they are not explicitly placed in any of the sub-teams.

Until this point, the sub-team analysis has been based entirely on the technical process analysis. The grouping of disciplines, however, leads to a People recommendation rather than a Process one because of the observed correlation between team coordination and team learning. This relationship indicates that the team learns the most when their reported interactions follow the patterns shown in Figure 7-1. This implies that the people on the team can work more productively if they follow the sub-team divisions that result from clustering the network of team interactions. This leads to the next recommendation, which is

R2: ***Organize the team into sub-teams according to the team-based DSM. Although these sub-teams cannot work in isolation from each other, they are highly internally interdependent.***

Because the analysis of shared knowledge presented in this thesis is done in aggregate, it does not lead to any specific recommendations about the staffing of the MDL, i.e., which people should staff each role. Still, this type of recommendation could eventually be made based on some of the future work that follows from this research. Recall from Chapter 5 that the structural approach to shared knowledge is captured in the structure of the shared mental model network at a point in time. Although the present analysis focuses on an overall structural comparison of pre-session and post-session networks, a variety of social network analysis techniques could be applied to each network. This type of analysis may yield results pertaining to individuals, such as common roles of a given design team member across sessions and/or similarities and differences between the relative positioning of design team and customer team members in the shared mental model network. For the purposes of this thesis, however, the recommendations deal with the structure and organization of the team as a whole.

7.2.2. Process

The recommendations pertaining to the MDL process are drawn directly from the results of the analysis of the parameter-based DSM presented in Chapter 4. Partitioning the DSM revealed the phases of the design life cycle and a large tightly coupled block corresponding to the Engineering Design Phase. This form of the DSM is useful for identifying the design budgets that serve as starting assumptions because they appear as horizontal strings of marks across the coupled block. Based on this insight, the next recommendation is

> **R3:** ***Determine the starting assumptions to the design process. Generally, design budgets and other similar collections of parameters received by many of the discipline engineers can be assigned values based on assumptions made from similar past systems at the start of the first design iteration. The data on past systems can be based on previous MDL sessions or on existing systems designed in other settings.***

Beyond that, the only insight for process improvement to be gained directly from the partitioned DSM alone is the timing of the sequential phases that occur before and after the Engineering Design Phase. Because these phases already comprise a straightforward part of the process, they are not included in the list of recommendations.

To truly optimize the process, the next step is to tear the design budgets from the DSM. As shown in Chapter 4, this reveals a nearly sequential design process with just one small coupled block, the Orbit Determination Phase. Once the trades involved in that block have been resolved, the remaining sequential process can be implemented in a straightforward manner. Since the work is based on a set of starting assumptions, the sequential process must then be iterated to refine those assumptions. The structure of the torn DSM makes it feasible to complete a pre-defined number of well-structured iterations. From the torn DSM, two distinct Process recommendations are reached. The first of these recommendations is

R4: ***Resolve the critical design trades within the small coupled block of the torn DSM, i.e., the Orbit Determination Phase. This trade focuses on orbit determination and the communications architecture. This phase of the process primarily involves Communications, Flight Dynamics, and Mission Operations, but the results also affect certain aspects of four other disciplines.***

The above recommendation, R4, should be implemented concurrently with R1 (the period of learning and consensus building) to ensure that the entire team receives ample exposure to the central Communications subsystem design. The importance of concurrency in implementing these two recommendations is highlighted by the role of Communications in the model of shared knowledge in the team. After that, the next recommendation to be carried out is

R5: ***Design sequentially… then iterate. The number of iterations and the duration of each can be based on the expected difficulty or complexity of the design as determined before or during the period of learning and consensus building. Alternatively, the number and duration of iterations can be based on customer preference or on other organizational constraints.***

To best leverage the advantages of a pre-defined and well-structured sequential process, the design should be automated to the greatest extent possible. A sophisticated set of software tools cannot replace the role of the people or obviate the need for collaboration in the design process, but with the proper technology helping to achieve the design objectives, the MDL can get the most benefit from both the expertise of the people and structure of the process. The next subsection addresses this with a recommendation related to the MDL Tools.

7.2.3. Tools

As discussed in Chapter 3, the MDL has created a variety of increasingly sophisticated software tools to facilitate data exchange (Karpati et al. 2003). These integrated system tools have eased the transfer of discipline outputs, but even the most current tool, the Process Reasoning and Information Management Environment (PRIME), does not capture the interdependencies across all disciplines in the team. Since it merely allows engineers to post their output parameters, PRIME cannot be used to facilitate design process automation. The DSM, however, captures all dependencies in the MDL process and therefore can be used as the basis for a data exchange tool that automates much of the information flow in the process.

As with many complex data sets, the catalogued dependencies can be managed in a relational database. Using this structure, a table or set of tables would be used to represent each discipline involved in the design process. As a notional example of how such a tool might be constructed, each table would contain the discipline's output parameters in the rows and input parameters needed from other disciplines in the columns. The content would be the actual values of the parameters being passed. Using the appropriate queries, each discipline engineer would be able to track the input parameters needed from other disciplines and the fate of the parameters that he or she provides as outputs.

Building from this database back-end, the MDL engineers and/or support staff could write a variety of applications to automate much of the process, and a web interface could be developed to quickly display the results of the critical design trades. In addition, as this database is updated and maintained across many design sessions, it would evolve to a point such that it can be used to automate the process of determining starting assumptions as discussed in recommendation R3. Once certain filters are applied to specify the general type of mission concept under study, the fields for the design budgets could be populated automatically with starting assumptions based on data from past design sessions. Given this opportunity, the next recommendation for the MDL is

R6: ***Develop a database-driven software tool for design process automation based on the parameter dependencies captured in the DSM. This recommendation is not meant to apply to each design session individually but rather as an infrastructural investment to improve the productivity of future design sessions.***

Of course, implementing the recommendations related to People, Process, and Tools is dependent on the Facility in which the work is conducted. The recommendation for the MDL Facility is discussed in the next subsection.

7.2.4. Facility

Since its inception, the Mission Design Laboratory has experimented several times with alternative arrangements of the discipline engineers' work stations in the room. As discussed in Chapter 3, some disciplines are seated near each other because the MDL management has deemed that a need exists for frequent interaction while others are placed near each other simply because certain work stations were available. Although the purpose of the MDL facility is to ensure frequent communication among the entire team, there undoubtedly is still value in determining a seating arrangement based on an analysis of design process data. As Allen (1985) has shown, the frequency of communication in research and development organizations is related inversely to the distance between engineers – a relationship that has become known as the Allen curve. Since this result is based on larger organizations requiring less frequent communication, the ICE example represents only a small range at the extreme of the curve. Still, Allen's findings suggest that the drop-off in interactions occurs quickly at relatively short distances. Thus, the Allen curve provides some support for the importance of identifying sub-teams of highly interdependent disciplines in the MDL.

Recall the use of the clustered team-based DSM for determining sub-teams in the MDL. In addition to this application, the team-based DSM can also be used to improve the seating arrangement in the facility. This analysis, however, does not require that the DSM be clustered. Instead, it is based on particular pairs of disciplines that are interconnected. The seating arrangement could be chosen in a number of ways depending on the philosophy employed by the

MDL management and/or the preferences of the customer team. For example, the work stations might be arranged to maximize the number of disciplines adjacent to other disciplines with which they are expected to interact. Ultimately, the layout of the facility should depend on a number of factors and could vary from one design session to the next. Making this type of change on a weekly basis is already feasible because each discipline engineer is able to log in at any of the work stations in the facility. Regardless of the exact seating arrangement used in a given session, though, the team-based DSM can serve as a guide for structuring the facility. Thus, the final recommendation for the MDL is

R7: ***Arrange the layout of the MDL facility to leverage the combined contributions of interdependent disciplines. The precise work station assignments can vary based on features of each design session, but the structure of the team-based DSM can serve to guide the choices made in each case.***

In sum, seven recommendations have been made to guide the MDL design process based on the analysis presented in this thesis. Two recommendations relate to People, three to Process, one to Tools, and one to Facility. The next section offers a standardized model of the ICE design process that incorporates all of these recommendations and several other aspects of the analysis presented in the thesis.

7.3. A Standardized Design Process Model for ICE

In this section, the insights of the entire thesis are consolidated into a single proposed model for standardizing the ICE design process. For the most part, this model is descriptive, but it has some prescriptive capacity that comes from its standardization of the existing process based on a formal analysis of empirical data. The standardization of the process is made possible by three aspects of the analysis. First, because the DSM was constructed to represent a typical design session, it can be used as a standard for how MDL design sessions normally proceed. Second, since the analysis of shared knowledge is based on a comparison of 12 different design sessions spanning most of the possible types of missions that could be encountered in the MDL, the results of that analysis can be applied to nearly any session based on its similarity to one or more of the observed sessions. Finally, the statistically significant correlation between team coordination and shared knowledge across all observed design sessions demonstrates the applicability of the DSM not only to the typical sessions for which it was constructed but to all MDL sessions.

The standardized design process model is shown in Figure 7-2. As the model depicts, the design process begins with a customer request. According to the partitioned DSM, the first step in the design process is the Requirements Definition Phase. According to recommendation R1, that phase is followed by a period of learning and consensus building, which helps to ensure that necessary changes in shared knowledge occur before the full design session commences. Concurrent with that period, the standardized model includes the resolution of orbit determination design trades in the core of interdependent disciplines as suggested by recommendation R4. Because these two steps both occur early in the process, they are coupled in the standardized process so that the design trades can be used to guide the consensus-building process (and vice versa). The coupling of these two steps is especially important because the Communications subsystem has been shown to be an indicator of the dynamics of shared knowledge in the team.

Based on the insights from the period of learning and consensus building, the next step is to establish the design budget estimates to be used as starting assumptions in the first iteration of the design (recommendation R3) and to determine the number and length of each iteration (recommendation R5). According to recommendation R6, the implementation of several design iterations is facilitated by a process automation software tool based on the dependencies in the large coupled block of the partitioned DSM (and that tool can also be structured to retain data from past sessions, which would eventually simplify the process of making the starting assumptions). According to recommendation R7, information flows freely where needed due to the strategic layout of the work stations determined from the team-based DSM. Finally, based on recommendation R2, sub-teams are formed around the groupings revealed by the team-based DSM. These sub-teams meet periodically during the Engineering Design Phase to ensure that actual interactions match expected interactions as documented in the team-based DSM.

The Engineering Design Phase is iterated a number of times specified in the previous step. Following each iteration of that phase, the design budgets are updated, refined, and used as

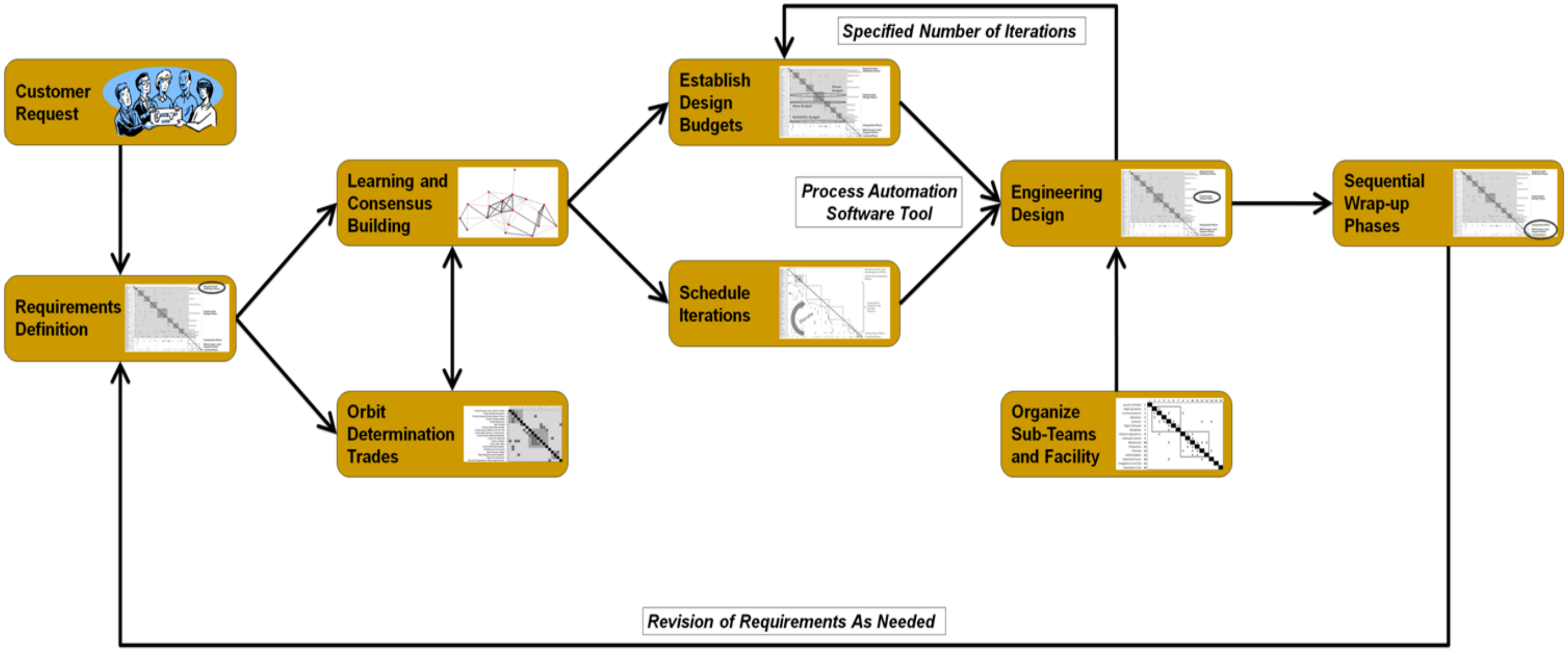


Figure 7-2. Standardized Design Process for the ICE Environment. Based on the results presented in this thesis, a standardized model of the ICE design process can be constructed. Some of the key elements of this model are a period of learning and consensus building, upfront resolution of orbit determination trades, starting assumptions for the design budgets, a specified number and length of design iterations, and the formation of sub-teams based on information flow for a typical design session.

inputs to the next iteration. After all iterations have been completed, the design proceeds to the Integration, Maintenance and Support, and Costing Phases. Finally, the results are compared to the initial requirements, and the entire process is iterated as needed, possibly in a newly commissioned design session, depending on the time and resources of both the MDL and the customer team.

Although the data have shown that this standardized process is applicable to all MDL design sessions, it cannot necessarily be applied as-is in other ICE design centers. Each of these centers has its own set of team roles, facilities, tools, level of customer interaction, and types of projects that they consider. Nevertheless, the model presented here provides a guideline for studying other ICE settings that can be modified and subsequently applied more generally. In addition, the model can also be used as a guide for planning the creation of new ICE design centers. This potential outcome of the research is the focus of the next section.

7.4. Suggestions for Establishing a New ICE Design Center

The creation of a new ICE design center involves a number of technical and organizational issues that fall outside the scope of the analysis presented in this thesis. These issues have been considered by people with a depth of personal experience at various design centers. For example, Joel Sercel of the California Institute of Technology has leveraged his experience working in ICE to create a firm called ICS Associates, offering consulting services aimed at the establishment of integrated concurrent design capabilities within client organizations.[27] In addition, a number of guides on organizational design more broadly have been written by experts in that field (e.g., Galbraith et al. 2002). The goal of this section is not to replace these resources. Instead, the purpose here is to briefly reframe the recommendations made in this chapter and to make some suggestions based on systematic and data-driven analysis of an existing ICE design center.

Given the value that DSM-based analysis can provide, the first step in creating a new ICE facility is to build a DSM representation for the expected typical work of that design center. This is not a trivial task and would require a large organization-wide effort to map all of the

---

[27] http://www.icsassociates.com/mission.htm

dependencies in the relevant process, but the reward would be significant and the analysis applicable outside of the new ICE laboratory as well. The DSM presented in this thesis can be used as a general guideline for this process, but it cannot be used as-is in most (or even any) other contexts. If the typical product of a potential design center is expected to be something other than an Earth-orbiting scientific spacecraft and the associated mission architecture, the precise parameters used and the dependencies among the parameters would differ. Once the DSM has been constructed, it can be analyzed using the procedures outlined in Chapter 4. This would aid in the development of the process for the new center, the starting assumptions to that process, and any remaining interdependent design trades about which assumptions cannot be made (like the core of interdependent disciplines for the MDL).

This analysis might even help to determine whether ICE is the appropriate setting in which to conduct the work in question. Recall that the partitioned DSM for the MDL contains a single large coupled block containing the actual engineering design work, which consists of 132 of the 172 total parameters. It would be reasonable to expect the DSM for other ICE design centers to have a similar structure because of the ubiquitous information flow among everyone involved in the process. Still, because the research is based on a single design center, one cannot assume that this structure is a necessary condition for any ICE laboratory. This reveals an important issue and an area of future work. If the structure of the partitioned DSM for the potential ICE center is significantly different from the one presented in this thesis, it might be worth considering other options for structuring the process. Further development of this problem would contribute to the literature on product architecture and organizational structure discussed in this thesis.

Assuming that the decision is made to proceed with the creation of an ICE center, the DSM loop analysis and resulting team-based DSM can be used in much the same way as for the MDL to identify critical design trades and interdependent disciplines. For a new design center, however, this analysis could do even more than that. Depending on the structure of the organization within which the new center would operate, the roles on the team can actually be determined based on this analysis. It is important to note, however, that changing team roles would probably not be an appropriate strategy for the MDL at this point for two reasons. First, the MDL already operates with certain team roles, and it would likely be a more drastic adjustment than anything else suggested in this chapter to change them. Second, many of the roles in the MDL are dictated by the organization of Goddard Space Flight Center (GSFC) because each discipline engineer is assigned to his or her role by the GSFC Branch head. If a new center were to be structured differently, however, the DSM could be used in guiding the initial allocation and assignment of team roles for that center. Even if this is not possible, the team-based DSM can still be used to organize the layout of the facility and to identify sub-teams.

To apply the shared knowledge research to the development of a new ICE design center, surveys can be distributed to engineers working in the existing organizational structure. Although the full development life cycle is probably too long to make pre-project and post-project surveys feasible, it would be possible to distribute surveys at two intermediate times as an initial test of how shared knowledge changes over time. Depending on the results of this analysis, the manager of the new ICE center can decide if – or under what circumstances – the period of learning and consensus building suggested for the MDL should be incorporated into the process used in the new center. This might require distributing surveys in multiple projects, in which case this step could be completed only if the organization has several programs running at the same time.

The other direct use of the shared knowledge analysis is determining its relationship to socio-technical congruence as proposed in Chapter 6. This, of course, requires data on actual (or reported) interactions. This should be done for the entire organization on which the potential ICE center is based only if it is deemed to be of other use to that organization. Although an analysis of communication patterns in the organization would be valuable for a variety of purposes, it is rather resource-intensive and unlikely to be essential for initially establishing the new center. If this analysis were conducted, however, it could be done in the larger organization by tracking electronic correspondence (as done by Cataldo et al. 2008) or other types of communication. Given the resulting data set, along with the team-based DSM and the results of the shared knowledge work, the analysis could then be conducted in the same way as described in Chapter 6. If this work is not done, however, team interaction data can still be collected during the first few sessions of the new ICE center, and its operations can be adjusted accordingly based on the results. After all, this type of testing and readjustment is likely to occur during an initial period following the establishment of a new organization of any type, size, or scope.

7.5. Beyond ICE: Cognition and Process in Systems Engineering Organizations

This chapter has offered some insights and recommendations regarding the future of the Integrated Concurrent Engineering design environment. The results presented in this thesis are directly applicable only to the ICE design center on which the research is based. The suggestions made in the previous section are meant specifically to guide the development of new ICE laboratories. At the same time, though, many aspects of the methodology can be applied to larger organizations. In the previous section, it was recommended that much of the methodology be applied in a restricted manner to a full organization to guide the structure of a potential new ICE center. This section expands on that discussion by suggesting ways in which the methodology can be applied directly to those organizations.

The DSM analysis used in this thesis is directly applicable to large organizations because the methodology has generally been applied in that type of setting. In fact, this thesis represents the first application of the DSM methodology to a small rapid design environment. The question that remains, then, is whether the actual DSM constructed for the ICE environment is applicable to a full space systems development program. Because the team roles in the MDL are representative of a full program, it should be feasible in principle to transfer the DSM to that context. Nevertheless, in a full development program, a task-based DSM might be more appropriate than the parameter-based one used in this thesis to represent conceptual design, though this DSM can still be used as the basis for constructing a similar task-based DSM. In addition, the DSM built for a typical MDL design session cannot work as-is outside of the MDL. Every program is different, and so it would not be possible to construct any single DSM that can be applied to all programs. Recall that one of the advantages of the ICE environment is that it facilitates the construction of a DSM for an entire space mission, which could be prohibitively complex for the construction of a full DSM. The DSM presented in this research, however, provides an accessible way to *begin* the construction of a DSM for a full program. Using this DSM as a baseline, program managers can make the necessary additions and adjustments to create a DSM that is directly applicable to their particular program.

One way of scaling the DSM could be to split it into 16 separate DSMs, the blocks along the diagonal of the unprocessed DSM shown in Figure 4-2. Each of these probably would be expanded to capture the complexity of the organizations that develop the individual subsystems and other technologies. Following that, the interconnections in the DSM for ICE could be used to reintegrate the 16 DSMs back into a single large one. The analysis discussed in Chapter 4 could then be repeated on each of the separate discipline DSMs, on the entire reconstructed DSM, or on both. The particular insights, of course, can be expected to differ, but that is true for other ICE design centers as well. Of course, in the implementation, the work would be drastically different. In a full space systems development organization, a larger number of iterations would be required, and the team-based DSM could be used to determine organizational structure rather than sub-teams of a few people. For these reasons, application of the DSM to larger programs could provide even greater benefit than it does to ICE.

The shared knowledge research presented in Chapter 5 also can be applied to larger organizations. Unlike the DSM, research on shared knowledge originated in small teams, and the work in that area has not expanded significantly to large organizations. Of course, extensive bodies of literature on organizational culture and institutional memory exist, and some of this work explicitly addresses the role of cognition in organizations (e.g., Meindl et al. 1996). This work might be akin to the team mind that is part of the naturalistic approach or even team cognition as described by the holistic approach. Still, the construct of shared mental models, the basic building block not only of the collective approach but also of the structural approach proposed in this thesis, has not been applied to large organizations to any notable extent.

Using the structural approach, it would be possible to apply research on shared mental models to organizations of any size. One of the significant advantages of network analysis is its scalability. Applying the analysis to a larger organization is merely a matter of adding more nodes and edges to the network. This makes some of the analysis more difficult, but it also creates an opportunity for other analysis that relies on large sample sizes and thus might not be applicable to the smaller shared mental model networks presented in this thesis.

A larger engineering organization most likely would have a different and much larger set of possible design drivers from which to choose, but the basic computation of a shared mental model can be done in the same way as demonstrated in the thesis. On the other hand, if an organization's goals are drastically different from those of the MDL, the possible drivers offered on the surveys or even the entire definition of what constitutes a pair-wise shared mental model can be adjusted accordingly. In that case, the measure of mental model sharedness would be different, but the basic network structure could still be used to analyze shared knowledge throughout the team. In addition, the longer time scales of the projects would facilitate data collection at many points in time so that a full analysis of the evolution of shared knowledge over time could be conducted. Finally, in an organization in which time scales are longer and communication more deliberate, it would be more feasible to collect data on the actual (as opposed to reported) flow of information in the process and thus to determine the relationship between those interactions and shared knowledge in the organization.

The next and final chapter extends the analysis of this section by considering the broader implications of the research. It returns to and answers the research questions presented in Chapter 1, and it offers a means of expanding the existing definitions of systems engineering by incorporating the thoughts and views of the engineers designing the system. Whereas this chapter has offered the practical implications of the research, the next chapter synthesizes all aspects of the thesis – both applied and theoretical – to present a coherent framing of the contributions and the directions of new research yet to be explored.

Chapter 8

Conclusions and Future Work

This thesis has presented an interdisciplinary socio-cognitive examination of the design of a particular complex engineered system. Not only does the research offer a systems-level analysis of the full space mission design process based on an aggregative model of parameter dependencies, but it also offers an analysis of shared knowledge among the engineers designing the system. Most importantly, it integrates these two analyses to provide a complete picture of how systems engineering (SE) is actually done. Although the immediate application of the research is strictly to the Mission Design Laboratory (MDL) or at most to the Integrated Concurrent Engineering (ICE) design environment in general, the methodology developed in the thesis is extendable and generalizable to a variety of design settings.

The purpose of this chapter is to summarize and synthesize the results and contributions of the thesis. In section 8.1, the three research questions that were posed in Chapter 1 are answered. Then, section 8.2 offers a new definition of systems engineering that builds on existing definitions by incorporating the results of this thesis. Sections 8.3 and 8.4 explain the contributions of the research to the academic literature. The first of these two sections focuses on the literature reviewed in Chapter 2, while the second explains the place of this thesis within the growing body of work in the field of Engineering Systems. After that, section 8.5 discusses the important limitations of this research. Finally, section 8.6 offers several areas for future work to expand the impact of the research that this thesis has only just begun.

8.1. Research Questions Revisited

Recall that this research is divided into the three parts: (1) an analysis of the design process, (2) a model of shared knowledge in the design team, and (3) an integrative study connecting technical information flow to shared knowledge. The three questions on which the research is based are framed around these three components of the work. The purpose of this section is to revisit the research questions and to provide answers to each. These answers are presented in an encapsulated form that is meant to summarize the results of the work and thus do not capture the nuance of the analysis of the preceding chapters. Still, they offer a quick and accessible reference to the primary aims and outcomes of the thesis.

The first research question deals with the technical design process and the Design Structure Matrix (DSM) in particular. The technical research question is

Q1: ***How can the Design Structure Matrix be used to analyze and improve the process in a rapid collaborative design environment?***

This research question was addressed in Chapter 4, which demonstrated a full analysis of the design process in the ICE environment. Based on this work, the answer to research question 1 is

A1: ***Provided that it is constructed according to three guiding principles that account for the ubiquitous information flow in the ICE environment, the DSM can be used to map the phases of the design life cycle, identify critical design trades and interdependent disciplines, and determine the set of starting assumptions that, if made, optimize the process.***

The second question deals with the social dimension of the design team. The social research question is

Q2: ***How can a network-based approach reveal the dynamics of shared knowledge in engineering design teams?***

This research question was addressed in Chapter 5 with the development of a quantitative, scalable, and dynamic model of shared knowledge in teams. According to this portion of the thesis, the answer to research question 2 is

A2: ***A network-based approach can reveal the dynamics of shared knowledge in engineering design teams by integrating the advantages of the naturalistic, collective, and holistic approaches to reveal the structure of shared knowledge. A comparison of this structure at different points in time leads to a metric of change in shared knowledge that varies with technical attributes of the system being designed.***

The third and final question captures the interdisciplinary component of the thesis. The socio-technical research question is

Q3: ***What is the relationship between the design process and shared knowledge in engineering systems design?***

This research question was addressed in Chapter 6, which presented an integrated analysis of the previous two chapters and provided insights that can only come from an interdisciplinary perspective. According to this portion of the thesis, the answer to research question 3 is

A3: ***Team coordination and the design product are both closely related to the dynamics of shared knowledge in the team, but they are not necessarily directly related to each other. Additionally, certain aspects of the system at the "core" of the design process may serve as indicators of shared knowledge in the team.***

In the MDL design process, the aspects at the "core" of the design are the location of the spacecraft and its means of communicating with the ground. Applying the methodology presented in this thesis to the design of other systems may or may not demonstrate a similar role for certain central aspects of the process in those contexts.

The direct answers to the research questions presented above offer a concise set of insights that follow from the thesis, but they can also mask the richness and specificity of the analysis developed in the preceding chapters. Thus, these answers are not intended as a set of ready-made solutions to be applied in any design environment but rather as a reference that broadly describes the results of the research. As future work is completed in this area, the answers to these questions can be refined and adjusted based on the outcomes of that research.

8.2. Systems Engineering Redefined

One of the central goals of this thesis is to contribute to the ongoing debate regarding the definition of systems engineering, and the purpose of this section is to offer a new definition of the term. The goal here is not to replace existing definitions but rather to extend and strengthen them through the inclusion of the system's designers and developers as stakeholders. The creation of this definition is not merely pedagogical in its intent or in its result. Instead, it is a substantive contribution to the theory and practice of systems engineering that, hopefully, will be adopted as *part* of the overall picture of what SE is and how it is done. This new definition is:

> *Systems engineering is a socio-technical practice characterized by the creation and execution of an iterative process in which the individual knowledge, thoughts, and viewpoints of a diverse set of professionals combine and converge toward a design solution that delivers value to all stakeholders, including the customers, the users, and the designers.*

The definitions reviewed in Chapter 2 each presented a different perspective on systems engineering, each using one or two words to describe "what" SE is. These words, used alternately throughout the definitions, included "discipline," "process," "technology," "art," and "science." One definition, in an apparent attempt to be more specific than the others, refers to systems engineering as a "combination of theories and tools, carried out through use of a suitable methodology and set of systems management procedures" (Sage 1992, p. 10).

The definition proposed here comes from a broader perspective on systems engineering. The word "practice" is used to refer to any of a number of possible activities by which the result of a project is reached. It does not imply a mandate of what the final system should look like but rather a bottom-up emergence of a design based on the contributions of all individuals involved. Although the customer provides the requirements and the systems engineer the direction, the actual design represents a convergence of shared knowledge among the members of the team. As this research has shown, shared knowledge in the team converges over time, especially during the design of the most complex systems. Furthermore, the analysis indicates that change in shared knowledge increases with the level of team coordination. These findings motivate the perspective that SE results in a convergence of shared knowledge toward a design product. In addition, it supports the notion that the engineers, like the customers and the users, are people with individual preferences and thus should be viewed as one of the key stakeholder groups. This connection of people to the product and the process makes the practice a "socio-technical" one.

Lastly, it should be noted that of all the descriptors used in the previous definitions of systems engineering, the one proposed here retains the word "process." In contrast to the others, however, this definition highlights that it is an *iterative* process. Without iteration of the design and the frequent give-and-take among the engineers and the customers, it is unlikely that the convergence of knowledge and views could occur. Thus, iteration is a critical feature of the design process, but it is still just one aspect of a broader phenomenon that arises from the interdisciplinary and integrative work of many individuals. Furthermore, the need to both create and execute this process is retained from the INCOSE definition of SE (INCOSE Communications Committee 2006) because it highlights the necessary differences among various systems that contribute to the importance and the complexity of systems engineering. And for this reason, no single definition of systems engineering, including the one presented here, can adequately capture all aspects of SE in all contexts. As stated above, the proposed definition is meant only to enrich the existing definitions and to contribute to the overall theory and practice of systems engineering.

8.3. Contributions to the Literature

The purpose of this section is to provide a brief high-level overview of how this thesis has built on previous research and contributed to the total body of knowledge in three different areas. Following the format of the literature synthesis presented in section 2.5, this section is similarly divided into contributions to the three areas: design process analysis, theories of shared knowledge and cognition, and the intersection between the technical and the social.

8.3.1. Design Process Analysis

The technical aspect of this research has contributed to design process analysis at several levels. In one sense, the DSM analysis represents an application of an existing methodology to a new problem. The new context to which the DSM is applied can be seen either as the ICE environment or as space mission design. In the former case, a contribution to the DSM literature came in the form of three guiding principles that facilitate systematic analysis of information flow in an environment characterized by such ubiquitous communication. In the latter, the contribution is to the space systems literature. It is a means of constructing a baseline DSM for the extremely complex space mission design process in a closed environment in which such a prohibitively resource-intensive task otherwise would not be possible.

In another sense, the contribution is to the set of specific ways of using the DSM. For example, a method for converting a parameter-based DSM to a team-based DSM using loop analysis was devised for this thesis. This method can be viewed as a substantive contribution to either the DSM or to space systems, but the method's direct applicability to other contexts and the general applicability of the results to space systems design remain open questions. In addition, this research has used the DSM formalism to build on the relatively new topic of socio-technical congruence (STC). Specifically, this research proposes a new metric for STC that incorporates lean principles and the imperative to eliminate waste.

Finally, the thesis has also made important contributions to the literature on ICE. Those contributions, however, were discussed at length in Chapter 7 and thus do not need to be repeated here.

8.3.2. Shared Knowledge and Cognition

The primary contribution from the social component of this research is a rather significant one, but it can be stated simply. In short, this thesis has offered a fundamentally new approach to the analysis of shared knowledge in teams. Before doing this, the thesis first made a smaller but still important contribution in terms of framing the problem. Expanding on the categorization created by Cooke and Gorman (2006) of the collective and holistic approaches to shared knowledge, this thesis has restructured that scheme by combining it with another extant perspective on shared knowledge – the naturalistic approach.

With this framework established, the thesis explained how the advantages of the naturalistic, collective, and holistic approaches could be combined into a new approach, proposed in this thesis, called the structural approach to shared knowledge. By constructing a network of shared knowledge that is built up from pair-wise shared mental models, it is possible to quantitatively examine shared knowledge in teams of any size. Furthermore, the approach's scalability gives it the capacity to be applied to any real-world environment in which shared mental models in dyads can somehow be measured. Finally, a structural comparison of the network at two points in time leads to a metric for change in shared knowledge that was shown to vary with technical attributes of the system being designed. This quantification of the dynamics of shared knowledge constitutes a significant contribution both in itself and in combination with the proposal of a new approach to the study of shared knowledge in teams.

8.3.3. Socio-Cognitive Analysis of Engineering Systems Design

Broadly, this research has contributed to existing knowledge through its explicit connection of the social to the technical. Specifically, it has shown that the notion of shared knowledge in teams is, as previously suggested by Badke-Schaub et al. (2007), applicable to engineering design. This has led to conclusions about the relationship between shared knowledge and various technical attributes of the design product. The research has also shown that certain disciplines (in this case, Communications) may serve as indicators of shared knowledge. Of course, these insights apply only to the specific context of this research, but the socio-technical connection that has been revealed exists in the design, development, and use of any complex engineered system. To explore the contributions of the research in this area, the next section discusses the implications for a relatively new body of work focused on such engineered systems.

8.4. Implications for Engineering Systems

Engineering Systems is an emerging field whose purpose is to explicitly explore the interconnections between the technical and the social in the design, development, and use of complex engineered systems.[28] Thus, the three-pronged approach (technical, social, and socio-technical) taken in this thesis makes an important contribution to the growing body of literature in this field. The purpose of this section is to identify the position of this thesis among previous doctoral dissertations written in the Engineering Division (ESD) at the Massachusetts Institute of Technology.

Several doctoral theses completed in ESD have examined product development processes. Many of these works focused on delivering value to the stakeholders of the system (Downen 2005, McConnell 2007, Mostashari 2005, Ross 2006) while others were concerned more with developing methodologies for modeling the design and development process (Koo 2005, Smaling 2005, Suh 2005). Some theses, like the present one, specifically used the Design Structure Matrix methodology because of its ability to model processes from a systems perspective based on information obtained at the basic "nuts-and-bolts" engineering level (Bartolomei 2007, Browning 1998, Kalligeros 2006).

Many other theses, on the other hand, have explored the role of the organization. Haddad (2008) examined aspects of the link between product architecture and organizational structure through the mechanism of knowledge integration. Osorio-Urzúa (2007) built both on that area of work and on design process modeling in his exploration of how and why system architectures evolve over time. Building on prior work on X-teams, Stanke (2006) examined the role of groups of organizations and developed a theory of "X-enterprises." Hsieh (2008) used network analysis to examine the structure the Internet based on the evolution of standards in the system.

The present thesis contributes to several areas of ESD-relevant research, especially design process modeling, organizational structure, and the intersection between the two. Other ESD theses, however, have also explored some of the social aspects discussed in this research. One important ESD-relevant research area to which this thesis contributes directly is stakeholder alignment and group decision making (Lawson 2008, McKenna 2006, Tang 2006). One past thesis explored a similar problem to that of the current thesis but in a different way. Based on interviews with a large number of systems engineers working at a variety of levels within their organizations, Davidz (2006) developed a theory of how "systems thinking" develops in senior

---

[28] See http://www.cesun.org/ and http://esd.mit.edu/

systems engineers. Her thesis and the present one together address an important issue related to the role and purpose of systems engineering in organizations.

Finally, the present thesis breaks entirely new ground within the field of Engineering Systems with its analysis of the role of shared knowledge and cognition in teams. This is an important area of research that is beginning to receive attention in the Engineering Systems community. This thesis is just the first of many expected in the next few years that will explicitly address the role of knowledge and cognition in the design and use of complex socio-technical systems.

8.5. Limitations

The limitations of this research fall into three categories: organizational/contextual limitations, data limitations, and methodological limitations. The next three subsections address each of these in turn.

8.5.1. Organizational/Contextual Limitations

One of the first and most obvious limitations of this research is also one of its greatest strengths – its implementation in a single ICE design facility. Because all of the work was conducted in one setting, it was possible to compare the results of all observed design sessions directly to each other and to control for organizational issues that inevitably would be different across design centers. At the same time, however, this choice means that the direct applicability of the results obtained in the research is limited to the Mission Design Laboratory. Although the methodology has been tested and shown to be useful, the specific recommendations that come from the research could not be applied to any other context without repeating the data collection and analysis in those settings.

Another organizational/contextual limitation is that the context of the research – both the ICE environment and space mission design in general – exists within a rich historical and political backdrop. In many ways, this research assumes that the MDL operates in a vacuum. This, of course, is not the case. On one level, the MDL has been conducting design sessions since 1997, but this research is based only on a cross-section of 12 design sessions conducted between June 2007 and February 2008. The history of the MDL up to that point was not considered in the data collection or the analysis.

More broadly, ICE is only one of many paradigms of space systems design that have been implemented since the beginning of the space age. From the initial missions of the late 1950s to the Mercury, Gemini, and Apollo programs to the Space Shuttle and the wide variety of scientific, commercial, and defense satellites that have been designed over the past several decades, the structure of the design process has taken many forms. Although this research is the first analysis of shared knowledge in the ICE design setting, it is not the first formal study on the role of people in space systems design (e.g. Frischmuth and Allen 1969). Therefore, when viewed within the broader context of the entire space program, this research can be seen only as a contribution to a broad body of theoretical and practical work. Still, within the domain of shared knowledge and design process, the thesis opens an important area of interdisciplinary research for the future.

8.5.2. Data Limitations

The data used in this research come in four categories: interview data on parameter dependencies, survey data on major design drivers, survey data on reported team interactions, and documentation of system attributes for each design session after completion. The interview data on parameter dependencies has a few important limitations that should be noted. First, a Design Structure Matrix can be constructed in a more detailed and refined way than was done in this research. In general, it is possible to place in each cell of the DSM a variety of different values representing types and strengths of dependencies. In this work, however, the DSM was constructed with only a single type of mark in each cell indicating that a dependency exists. Second, the DSM was constructed almost entirely on the basis of interviews with the discipline engineers. It is a true representation of the technical design process in the sense that it catalogues the tacit technical knowledge among all of the engineers on the team and aggregates them into a systems-wide representation. Still, as with most DSMs (which are usually constructed using a similar interview process), it has the limitation that it is a representation of the system only to the extent that the technical knowledge of the engineers is representative of that system. Finally, the parameter-based DSM serves as a useful baseline for application of the DSM to other space mission design and development contexts, but a task-based DSM would probably be more appropriate when extending the work to full development programs.

The primary limitation of the design driver data is that these drivers represent only one of many possible ways of measuring shared mental models. Furthermore, the respondents were asked merely to check boxes to indicate their views. A more refined analysis would have been possible if a Likert scale were used to determine the strengths of those views. In addition, the possible design drivers chosen were relatively broad. They were, for the most part, the disciplines involved in the process. Although this is a reasonable set of possible design drivers, they cannot possibly fully depict any engineer's knowledge and cognition relative to the system being designed.

The main limitation of the team interaction data is that they consisted of surveys of self-reported interactions by the design team members. Each engineer was asked to rate the importance of communication with each other discipline in the process. This was done because measuring actual interactions among design team members in the ICE environment would have been a painstaking and laborious task in itself and quite possibly could have resulted in a completely populated interaction matrix in which everyone interacts with everyone else. The survey question on reported interactions, on the other hand, was designed to identify the most important instances of team communication. This might have been more accurately measured by making audio and/or video recordings of all interactions to determine which were the most important. Of course, this methodology is fraught with its own limitations as well.

The limitations of the data on system attributes are generally related to uncertainty in their measurement. Although the values used for system development time, launch mass, and mission cost are based on the final outcomes of the design sessions, those results are merely estimates from work done at an early conceptual design phase. Furthermore, the measure of Mission Technology Readiness Level (MTRL) is based on the judgment of the engineers and thus is subject to even greater uncertainty. Moreover, the extent of the uncertainty is also unknown because an estimate of error is not made for those metrics during the design sessions. Nevertheless, this limitation comes from a necessary tradeoff with data availability. Because the research was based on one-week design sessions, it was possible to collect data on the design of 12 distinct mission concepts. This would not have been possible if more detailed design and/or development programs had been used for the research.

Finally, the last major data limitation is also part of the system attributes category. This limitation is that a metric for quality of the final design product (or a related metric of team performance) was not available for the design sessions. Like the other data limitations, this is a result of the early conceptual phase of the design sessions. An attempt was made to measure design quality by asking the customer team about their expectations on the pre-session surveys and their satisfaction with the results on the post-session surveys. These open-ended responses, however, were inconclusive because the customers almost always reported that they were generally satisfied with the results.

To mitigate this limitation in future research, it might be more useful to offer a Likert scale including many levels of satisfaction from which to choose. This way, it could be possible to measure nuanced differences in the customers' views of the results. Additionally, design quality might be better assessed by introducing the concept of parallel strategies to ICE design sessions (see Abernathy and Rosenbloom 1969, Abernathy 1971, Frischmuth and Allen 1969). If sufficient resources are available, several separate ICE teams could conduct the same design session. This could provide an important dual benefit. It might lead to better design outcomes for the customer team, and it also could result in a measure (at least a relative one) of the quality of each design session's outcome. This measure could then be used to assess the relationship between change in shared knowledge and the performance of the team.

8.5.3. Methodological Limitations

The methodological limitations can be divided into two types that are based on the two general methodologies used: DSM analysis and measurement of shared mental models. DSM analysis consists of three general procedures: partitioning, tearing, and clustering. For the purposes of this thesis, a loop analysis method was also introduced. The partitioning and clustering analyses were done using standard software tools implementing those procedures, so the particular new limitations here are in the tearing and the loop analysis. The main limitation in both of these parts of the analysis is the same – that they were based largely on domain knowledge, visual pattern recognition, and the judgment of the researcher. Because of the highly complex and interdependent nature of the parameter-based DSM, however, this approach was necessary to make the work feasible. Overcoming this limitation in the future would require more sophisticated software tools and even more powerful hardware than are generally available to most researchers.

The measurement of shared mental models comes with a variety of limitations, some of which were discussed as data limitations in the previous subsections. Strictly methodologically, the important limitations are (1) the measurement of time dependence and (2) the lack of results on causal relationships. First, the change in shared knowledge measured in this research is merely a change from the beginning to the end of the work. If surveys had been distributed at multiple points throughout each design session (perhaps twice per day throughout each five-day session), it would have been possible to construct graphs depicting the time evolution of shared knowledge in the team. Thus, the results would have included not only the observation of a change in shared knowledge but also the exact profile of that change (e.g. linear, U-shaped, exponential, approaching some asymptotic limit, or discontinuous at a point in time). Second, the results in this thesis relating various features of the product and the process (change in shared knowledge, socio-technical congruence, perceived importance of drivers, technological maturity, launch mass, etc.) are all correlations. While they provide important insight about the relationships among shared knowledge, process, and product, the correlations used do not imply anything about causality. A more complete description of the causal relationships among the variables studied would require additional research. The next section provides an overview of the future work that can follow from this thesis, including addressing many of the limitations discussed above.

8.6. Future Work

To echo the final thought of the first chapter, the results of this thesis are exciting and promising, but they only scratch the surface of possible ground-breaking research in the socio-cognitive analysis of engineering systems design. Thus, the goal of this final section of the thesis is to explore some of the potential areas to which this research can be applied and ways in which it can be expanded.

First, before the methodology developed here and applied to the Mission Design Laboratory can be used in other contexts, it first should be tested in other ICE design settings. The natural next step for such analysis is the MDL's partner facility, the Instrument Design

Laboratory (IDL), also part of the Integrated Design Center (IDC) at NASA Goddard Space Flight Center (GSFC). The first step in this follow-on research has already been done. The surveys distributed to the MDL design team before and after each design session have been modified to apply to the IDL, and they were distributed to and completed by the design team during one IDL session. Although the results for a single design session cannot be used to demonstrate any trends in the IDL, the groundwork has been completed for this next step in the research to begin.

Similarly, the methodology proposed in this thesis can also be applied to other ICE design centers at other locations, such as Team X at the Jet Propulsion Laboratory (JPL). Not only can the DSM-based approach be applied directly to the Team X design process, but it can also be integrated with a multiagent simulation algorithm developed by Olson et al. (2009). The DSM provides both greater detail in the information dependencies and a systems-level view from which to evaluate the algorithm, while the simulation approach offers a means of determining the outcomes of certain design choices and could improve the accuracy and robustness of the expected interaction matrix used in this research.

In addition to the above design process analysis work conducted in the Team X context, Maria Yang of ESD and colleagues have examined speech patterns of Team X engineers in a series of studies that essentially constitute an initial application of the holistic approach to shared knowledge in the engineering design context (Ji et al. 2007, Yang and Ji 2007). This creates a valuable opportunity both to apply Yang's methodology to the MDL (and the IDL) and to continue her work with Team X by applying the methodology proposed in this thesis to that design setting. Following that, the research can be extended to any of the other ICE design centers discussed in Chapter 3. Furthermore, as described in Chapter 7, the methodology can also be expanded under certain conditions to be applied to full-scale development programs and other large organizations.

Methodologically, there are several areas of future research that can continue the design process work presented here. As discussed in Chapter 2, a Design Structure Matrix can be constructed with various values in its cells to represent strengths and/or types of interactions. In the setting in which this research was conducted, it was not possible to achieve that level of specificity in the data with confidence. A DSM using more refined data on parameter dependencies but otherwise applied as presented in this thesis would constitute an important new area of research. In addition, the means of determining the starting assumptions from the partitioned DSM was done by a combination of visual inspection of the DSM and the researcher's domain knowledge in space systems design. However, algorithms do exist to optimize the choices made in tearing (Gebala and Eppinger 1991). A more formal analysis of process optimization by tearing the DSM for the ICE process would be a valuable next step to pursue. Finally, the loop analysis of the coupled block of the DSM is an especially important area of future work. In principle, it is possible to complete a more purely algorithmic approach to loop analysis that does not simply establish a cutoff for loop length. Still, this would be extremely difficult because the computational power required to analyze all of the loops in such a large and tightly coupled network are not readily available.

In the area of shared knowledge and cognition, this research has made important methodological contributions. Since the work presented here is an entirely new approach to the problem, there are several areas of potential future work. First and most obviously, the structural approach to shared knowledge must be implemented in other settings to establish its general applicability. This can be done by repeating essentially the same work in other types of environments, but it also could mean making certain changes to the data collection and/or analysis. For example, the dyadic shared mental models (i.e., the edge weights in the network) can be measured in a number of different ways, or a different means of structural comparison between networks other than the quadratic assignment procedure (QAP) could be tested.

Furthermore, this thesis has not provided a true time series analysis of the evolution of shared knowledge because only two points in time were used for each design session. If the surveys were distributed once or twice per day over the course of a five-day session, it would be possible to measure not only *if* a convergence of shared knowledge occurs but also *when* it occurs and, in combination with other types of analysis, even *how* it occurs. This type of time series analysis might be better done in environments other than ICE in which time scales are longer and thus more surveys can reasonably be distributed. At the other end of the spectrum, however, it might be better to study teams that complete their work within an even shorter time frame. This way, nearly controlled experiments could be conducted in which various features would be adjusted one at a time to facilitate the measurement of causality and not just of correlations as presented in this thesis.

Much of the literature on shared knowledge uses team performance as a key variable. In the present research, it was not possible to measure either team performance or the quality of design outcomes because of the nature of conceptual design – there is no single right answer. In future research, the experimental approach discussed above could be applied to other types of teams or even to design teams working on more well-defined and encapsulated tasks. This would provide a means of examining team performance as it relates to shared knowledge as measured according to the structural approach.

Also, the structural approach to shared knowledge does not necessarily require that the time element be incorporated as it has been in this thesis. Another way of using the structural approach for the analysis of shared knowledge is to apply various network analysis methods to a static network depicting the structure of shared knowledge at a snapshot in time. This would be a useful first step in applying the structural approach to longer-term space systems development programs whose time horizons are generally far too long to measure pre-work and post-work levels of shared knowledge in the organization.

Finally, this thesis opens the door for further analysis in the integrated and interdisciplinary socio-cognitive approach to the study of complex engineered systems. Similar analyses could be conducted using entirely different design process analysis methodologies and/or different approaches to shared knowledge. The results of this thesis have demonstrated perhaps more than anything else that this is a strong and fruitful area of research. With the completion of this thesis, the gates have opened for further work in all aspects of an increasingly important problem – how people think and how their thoughts affect and are affected by the design of complex engineered systems. And with that, the invitation is hereby extended for discussion, deliberation, and collaboration on future work in the socio-cognitive analysis of engineering systems design.

References

Abernathy, W.J. (1971). “Some Issues Concerning the Effectiveness of Parallel Strategies in R&D Projects.” *IEEE Transactions on Engineering Management* EM-18(3): 80-89.

Abernathy, W.J. and Rosenbloom, R.S. (1969). “Parallel Strategies in Development Projects.” *Management Science* 15(10): B486- B505.

Aguilar, J.A. and Dawdy, A. (2000). “Scope vs. Detail: The Teams of the Concept Design Center.” *Proceedings of the IEEE Aerospace Conference*. IEEE, Big Sky, MT, Vol. 1, pp. 465-481.

Ahmadi, R.H., Roemer, T.A., and Wang, R.H. (2001). “Structuring Product Development Processes.” *European Journal of Operational Research* 130(3): 539-558.

Allen, T.J. (1985). *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization*. Cambridge, MA: MIT Press.

Allen, T.J. and Henn, G.W. (2007). *The Organization and Architecture of Innovation: Managing the Flow of Technology*. Burlington, MA: Butterworth-Heinemann.

Ancona, D., Bresman, H., and Kaeufer, K. (2002). “The Comparative Advantage of X-Teams.” *MIT Sloan Management Review* 43(3): 33-39.

Ancona, D.G. and Caldwell, D.F. (1992). “Demography and Design: Predictors of New Product Team Performance.” *Organization Science* 3(3): 321–341.

Apgar, H., Bearden, D., and Wong, R. (1999). “Cost Modeling.” In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 783-820. El Segundo, CA: Microcosm.

Badke-Schaub, P., Neumann, A., Lauche, K., and Mohammed, S. (1994). “Mental Models in Design Teams: A Valid Approach to Performance in Design Collaboration?” *CoDesign* 3(1): 5-20.

Baldwin, C.Y. and K.B. Clark (2000). *Design Rules: The Power of Modularity, Vol. 1*. Cambridge, MA: MIT Press.

Baldwin, C.Y. and Clark, K.B. (1997). “Managing in an Age of Modularity.” *Harvard Business Review* 75(5): 84-93.

Baldwin, C.Y. and Clark, K.B. (2001). “Modularity after the Crash.” *Networks, Organizations and Markets Research Papers* 01-05, Harvard Business School.

Bandecchi, M., Melton, B., Gardini, B., and Ongaro, F. (2000). “The ESA/ESTEC Concurrent Design Facility.” *Proceedings of the 2nd European Systems Engineering Conference*. International Council On Systems Engineering, Munich, Germany, pp. 329-336.

Baron, R.M. and Kenny, D.A. (1986). “The Moderator-Mediator Variable Distinction in Social Psychological Research: Conceptual, Strategic, and Statistical Considerations.” *Journal of Personality and Social Psychology* 51(6): 1173-1182.

Bartolomei, J.E. (2007). "Qualitative Knowledge Construction for Engineering Systems: Extending the Design Structure Matrix Methodology in Scope and Procedure." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Blanchard, B.S. (2008). *System Engineering Management*, 4th ed. Hoboken, NJ: John Wiley & Sons.

Blanchard, B.S. and Fabrycky, W.J. (2006). *Systems Engineering and Analysis*. Upper Saddle River, NJ: Pearson Prentice Hall.

Bonner, B.L., Baumann, M.R., and Dalal, R.S. (2002). "The Effects of Member Expertise on Group Decision-Making and Performance." *Organizational Behavior and Human Decision Processes* 88: 719-736.

Borgatti, S.P. (2002). *NetDraw: Graph Visualization Software*. Harvard: Analytic Technologies.

Borgatti, S.P., Everett, M.G., and Freeman, L.C. (2002). *UCINET 6 for Windows: Software for Social Network Analysis*. Harvard: Analytic Technologies.

Browning, T.R. (2001). "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions." *IEEE Transactions on Engineering Management* 48(3): 292-306.

Browning, T.R. (1998). "Modeling and Analyzing Cost, Schedule, and Performance in Complex System Product Development." *Technology, Management, and Policy*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Browning, T.R. and Eppinger, S.D. (2002). "Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development." *IEEE Transactions on Engineering Management* 49(4): 428-442.

Buede, D.M. (2000). *The Engineering Design of Systems: Models and Methods*. New York, NY: John Wiley & Sons.

Cameron, B.G., Crawley, E.F., Loureiro, G., and Rebentisch, E.S. (1991). "Value Flow Mapping: Using Networks to Inform Stakeholder Analysis." *Acta Astronautica* 62: 324-333.

Cannon-Bowers, J.A., Salas, E., and Converse, S.A. (1993). "Shared Mental Models in Team Decision Making." In *Individual and Group Decision Making*, ed. N.J. Castellan Jr., pp. 221-246. Hillsdale, NJ: Erlbaum.

Carroll, J.S., Hatakenaka, S., and Rudolph, J.W. (2005). "Naturalistic Decision Making and Organizational Learning in Nuclear Power Plants: Negotiating Meaning between Managers and Problem Investigation Teams." *Organization Studies* 27(7): 1037-1057.

Cataldo, M., Herbsleb, J.D., and Carley, K.M. (2008). "Socio-Technical Congruence: A Framework for Assessing the Impact of Technical and Work Dependencies on Software Development Productivity." *Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement.* Association for Computing Machinery, Kaiserslautern, Germany, pp. 2-11.

Chelsom, J.V. (1994). “Concurrent Engineering Case Studies: Lessons from Ford Motor Company Experience.” In *Concurrent Engineering: Concepts, Implementation and Practice*, eds. C.S. Syan and U. Menon, pp. 25-48. London, UK: Chapman & Hall.

Cho, S.-H., Eppinger, S.D., and Massachusetts Institute of Technology. (2004). *DSM@MIT*. Cambridge, MA: Massachusetts Institute of Technology.

Coffee, T. (2006). “The Future of Integrated Concurrent Engineering in Spacecraft Design.” *The Lean Aerospace Initiative Working Paper Series*. Massachusetts Institute of Technology, Cambridge, MA.

Cooke, N.J. and Gorman, J.C. (2006). Assessment of Team Cognition. In *International Encyclopedia of Ergonomics and Human Factors*, 2nd ed., ed. P. Karwowski, pp. 270-275. UK: Taylor & Francis Ltd.

Davidz, H.L. (2006). “Enabling Systems Thinking to Accelerate the Development of Senior Systems Engineers.” *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Defense Systems Management College (2001). *Systems Engineering Fundamentals*.

Dietrich, F.J. and Davies, R.S. (1999). “Communications Architecture.” In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 533-586. El Segundo, CA: Microcosm.

Downen, T.D. (2005). “A Multi-Attribute Value Assessment Method for the Early Product Development Phase With Application to the Business Airplane Industry.” *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Eisner, H. (2008). *Essentials of Project and Systems Engineering Management*, 3rd ed. Hoboken, NJ: John Wiley & Sons.

Eppinger, S.D. (1991). “Model-based Approaches to Managing Concurrent Engineering.” *Journal of Engineering Design* 2(4): 283-290.

Eppinger, S.D., Whitney, D.E., and Gebala, D.A. (1992). “Organizing the Tasks in Complex Design Projects: Development of Tools to Represent Design Procedures.” *NSF Design and Manufacturing Systems Conference*. National Science Foundation, Atlanta, GA, pp. 301-309.

Eppinger, S.D., Whitney, D.E., Smith, R.P.. and Gebala, D. (1990). “Organizing the Tasks in Complex Design Projects.” *ASME Conference on Design Theory and Methodology.* American Society Of Mechanical Engineers, Chicago, IL, pp. 39-46.

Eppinger, S.D., Whitney, D.E., Smith, R.P., and Gebala, D.A. (1994). “A Model-Based Method for Organizing Tasks in Product Development.” *Research in Engineering Design* 6(1): 1-13.

Eterno, J.S. (1999). “Attitude Determination and Control.” In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 354-380. El Segundo, CA: Microcosm.

Frischmuth, D.S. and Allen, T.J. (1969). “A Model for the Description and Evaluation of Technical Problem Solving.” *IEEE Transactions on Engineering Management* EM-16(2): 58-64.

Galbraith, J.R. (1994). *Competing with Flexible Lateral Organizations*, 2nd ed. Reading, MA: Addison-Wesley.

Galbraith, J., Downey, D., and Kates, A. (2002). *Designing Dynamic Organizations: A Hands-On Guide for Leaders at All Levels*. New York, NY: AMACOM.

Gebala, D.A. and Eppinger, S.D. (1991). "Methods for Analyzing Design Procedures." *3rd International Conference on Design Theory and Methodology*. American Society of Mechanical Engineers, Miami, FL, pp. 227-233.

Gorman, J.C., Cooke, N.J., and Kiekel, P.A. (2004). "Dynamical Perspectives on Team Cognition." *Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting*. Human Factors and Ergonomics Society, New Orleans, LA, pp. 673-677.

Gough, K.M., Allen, B.D., and Amundsen, R.M. (2005) "Collaborative Mission Design at NASA Langley Research Center." *NASA Technical Reports Server* 20080009754.

Gruenfeld, D.H., Martorana, P.V., and Fan, E.T. (2000). "What Do Groups Learn from Their Worldliest Members? Direct and Indirect Influence in Dynamic Teams." *Organizational Behavior and Human Decision Processes* 82(1): 45-59.

Gulati, R.K. and Eppinger, S.D. (1996). "The Coupling of Product Architecture and Organizational Structure Decisions." *Sloan School of Management Working Paper*. Massachusetts Institute of Technology, Cambridge, MA.

Guo, F. and Gershenson, J. K. (2004). "A Comparison of Modular Product Design Methods on Improvement and Iteration." *Proceedings of the ASME Design Engineering Technical Conferences*, American Society Of Mechanical Engineers, Salt Lake City, UT.

Haddad, M.G. (2008). "Knowledge Integration for Problem Solving in the Development of Complex Aerospace Systems." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Heim, J.C., Parsons, K.K., Sepahban, S.F., and Evans, R.C. (1999). "TRW Process Improvements for Rapid Concept Designs." *Proceedings of the IEEE Aerospace Conference*. IEEE, Aspen, CO, Vol. 5, pp. 325-333.

Hölttä-Otto, K. and de Weck, O. (2007). "Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints." *Concurrent Engineering: Research and Applications* 15(2): 113-126.

Hollnagel, E. (2003). *Handbook of Cognitive Task Design*. Mahwah, NJ: Lawrence Erlbaum.

Hsieh, M.-H. (2007). "Standards as Interdependent Artifacts: the Case of the Internet." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

INCOSE Communications Committee (2006). "A Consensus of the INCOSE Fellows." *INCOSE: International Council on Systems Engineering*. Update 2 Oct 2006. View 3 Feb 2009.
<http://www.incose.org/practice/fellowsconsensus.aspx>.

Jackson, S.E. (1996). "The Consequences of Diversity in Multidisciplinary Work Teams." In *Handbook of Work Group Psychology*, ed. M.A. West, pp. 53-76. Chichester, UK: John Wiley & Sons.

Jackson, S.E., Brett, J.F., Sessa, V.I., Cooper, D.M., Julin, J.A., and Peyronnin, K. (1991). "Some Differences Make a Difference: Individual Dissimilarity and Group Heterogeneity as Correlates of Recruitment, Promotions, and Turnover." *Journal of Applied Psychology* 76(5): 675-689.

Jehn, K., G. Northcraft, and M. Neale. (1999). "Why Differences Make a Difference: A Field Study of Diversity, Conflict, and Performance in Work Groups." *Administrative Science Quarterly* 44(4): 741–763.

Ji, H., Yang, M.C., and Honda, T. (2007). "A Probabilistic Approach for Extracting Design Preferences from Design Team Discussion." *Proceedings of the International Design Engineering Technical Conferences & Computers and Information in Engineering Conference,* American Society of Mechanical Engineers, Las Vegas, NV, pp. 1-10.

Kalligeros, K., de Weck, O., de Neufville, R., and Luckins, A. (2006). "Platform Identification Using Design Structure Matrices." *Sixteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE)*. International Council On Systems Engineering, Orlando, FL.

Kalligeros, K. (2006). "Platforms and Real Options in Large-Scale Engineering Systems." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Kameda, T., Ohtsubo, Y., Takezawa, M. (1997). "Centrality in Sociocognitive Networks and Social Influence: An Illustration in a Group Decision-Making Context." *Journal of Personality and Social Psychology* 73(2):296-309.

Karpati, G., Martin, J., Steiner, M., and Reinhardt, K. (2003). "The Integrated Mission Design Center (IMDC) at NASA Goddard Space Flight Center." *Proceedings of the IEEE Aerospace Conference*. IEEE, Big Sky, MT, Vol. 8, pp. 8_3657- 8_3667.

Kiekel, P.A. and Cooke, N.J. (2004). "Human Factors Aspects of Team Cognition." In *The Handbook of Human Factors in Web Design*, eds. R.W. Proctor and K.L. Vu, pp. 90-103. Mahwah, NJ: Lawrence Erlbaum Associates.

Kirk, J. (2007). *Count Loops in a Graph*. Natick, MA: Matlab Central.

Kirkpatrick, D. (1999). "Telemetry, Tracking, and Command." In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 381-394. El Segundo, CA: Microcosm.

Klein, G.A. (1998). *Sources of Power: How People Make Decisions*. Cambridge, MA: MIT Press.

Klein, G. and Crandall, B.W. (1995). "The Role of Mental Simulation in Problem Solving and Decision Making." In *Local Applications of the Ecological Approach to Human-Machine Systems, Vol. 2*, eds. P. Hancock, J. Flach, J. Caird, and K. Vicente, pp. 324-358. Hillsdale, NJ: Lawrence Erlbaum Associates.

Klein, G. and Miller, T.E. (1999). "Distributed Planning Teams." *International Journal of Cognitive Ergonomics* 3(3): 203-222.

Klein, G., Ross, K.G., Moon, B.M., Klein, D.E., Hoffman, R.R. and Hollnagel, E. (2003). "Macrocognition." *IEEE Intelligent Systems* 18(3): 81-85.

Klimoski, R. and Mohammed, S. (1994). "Team Mental Model: Construct or Metaphor?" *Journal of Management* 20(2): 403-437.

Koo, H.-Y.B. (2005). “A Meta-language for Systems Architecting.” *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Krackhardt, D. (1987). “QAP Partialling as a Test of Spuriousness.” *Social Networks* 9(2): 171-186.

Krackhardt, D. and Hanson, J.R. (1993). “Informal Networks: The Company Behind the Chart.” *Harvard Business Review* 71(4): 104-111.

Langan-Fox, J., Anglim, J., and Wilson, J.R. (2004). “Mental Models, Team Mental Models, and Performance: Process, Development, and Future Directions.” *Human Factors and Ergonomics in Manufacturing* 14(4): 331-352.

Langlois, R.N. (2002). “Modularity in Technology and Organization.” *Journal of Economic Behavior & Organization* 49(1): 19-37.

Lawson, C.M. (2008). “Group Decision Making in a Prototype Engineering System: The Federal Open Market Committee.” *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Lee, T.-S. and Thomas, L.D. (2000). *Cost Growth Models for NASA’s Programs, Final Report*. Marshall Space Flight Center, Huntsville, AL.

Lim, B.-C. and Klein, K.J. (2006). “Team Mental Models and Team Performance: A Field Study of the Effects of Team Mental Model Similarity and Accuracy.” *Journal of Organizational Behavior* 27: 403-418.

Littlepage, G.E. and Mueller, A.L. (1997). “Recognition and Utilization of Expertise in Problem-Solving Groups: Expert Characteristics and Behavior.” *Group Dynamics: Theory, Research, and Practice* 1(4): 324-328.

Maier, M.W. and Rechtin, E. (2002). *The Art of Systems Architecting*, 2nd ed. Boca Raton, FL: CRC Press.

Mankins, J.C. (1995). “Technology Readiness Levels: A White Paper.” National Aeronautics and Space Administration.

March, J.G. (1991). “Exploration and Exploitation in Organizational Learning.” *Organization Science* 2(1): 71-87.

March, J.G. and Simon, H.A. (1993). *Organizations*, 2nd ed. Cambridge, MA: Blackwell.

Marks, M.A., Sabella, M.J., Burke, C.S., and Zaccaro, S.J. (2002). “The Impact of Cross-Training on Team Effectiveness.” *Journal of Applied Psychology* 87(1): 3-13.

Marks, M.A., Zaccaro, S.J., and Mathieu, J.E. (2000). “Performance Implications of Leader Briefings and Team-Interaction Training for Team Adaptation to Novel Environments.” *Journal of Applied Psychology* 85(6): 971-986.

Martin, J.N. (1997). *Systems Engineering Guidebook: A Process for Developing Systems and Products*. Boca Raton, FL: CRC Press.

Mathieu, J.E., Heffner, T.S., Goodwin, G.F., Cannon-Bowers, J.A., and Salas, E. (2005). "Scaling the Quality of Teammates' Mental Models: Equifinality and Normative Comparisons." *Journal of Organizational Behavior* 26: 37-56.

Mathieu, J.E., Heffner, T.S., Goodwin, G.F., Salas, E., and Cannon-Bowers, J.A. (2000). "The Influence of Shared Mental Models on Team Process and Performance." *Journal of Applied Psychology* 85(2): 273-283.

McConnell, J.B. (2007). "A Life-Cycle Flexibility Framework for Designing, Evaluating and Managing 'Complex' Real Options: Case Studies in Urban Transportation and Aircraft Systems." *Technology, Management and Policy*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

McCord, K.R. and Eppinger, S.D. (1993). "Managing the Integration Problem in Concurrent Engineering." *Sloan School of Management Working Paper*. Massachusetts Institute of Technology, Cambridge, MA.

McKenna, N. (2006). "The Micro-foundations of Alignment among Sponsors and Contractors on Large Engineering Projects." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

McManus, H.L. (2005). *Product Development Value Stream Mapping (PDVSM) Manual*, Release 1.0. Lean Aerospace Initiative, Massachusetts Institute of Technology, Cambridge, MA.

McManus, H., Haggerty, A., and Murman, E. (2005). "Lean Engineering: Doing the Right Thing Right." *Proceedings of the 1st International Conference on Innovation and Integration in Aerospace Sciences*. Centre of Excellence for Integrated Aircraft Technology, Queen's University Belfast, Belfast, UK, pp. 1-10.

McMordie, R.K. and Panetti, A. (1999). "Thermal." In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 428-458. El Segundo, CA: Microcosm.

Meindl, J.R., Stubbart, C., and Porac, J.F. (1996). *Cognition Within and Between Organizations*. Thousand Oaks, CA: SAGE.

Morelli, M.D., Eppinger, S.D., and Gulati, R.K. (1995). "Predicting Technical Communication in Product Development Organizations." *IEEE Transactions on Engineering Management* 42(3): 215-222.

Mostashari, A. (2005). "Stakeholder-Assisted Modeling and Policy Design Process for Engineering Systems." *Technology, Management and Policy*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Murman, E., Allen, T, Bozdogan, K., Cutcher-Gershenfeld, J., McManus, H., Nightingale, D., Rebentisch, E., Shields, T., Stahl, F., Walton, M., Warmkessel, J., Weiss, S., and Widnall, S. (2002). *Lean Enterprise Value: Insights from MIT's Lean Aerospace Initiative*. New York, NY: Palgrave.

National Aeronautics and Space Administration (2007). *Systems Engineering Handbook*. NASA/SP-2007-6105 Rev1.

Newman, M.E.J. (2003). "The Structure and Function of Complex Networks." *SIAM Review* 45: 167-256.

Newman, M.E.J., and Girvan, M. (2004). “Finding and Evaluating Community Structure in Networks.” *Physical Review* 69(2): 026113(15).

Olson, J., Cagan, J., and Kotovsky, K. (2009). “Unlocking Organizational Potential: A Computational Platform for Investigating Structural Interdependence in Design.” *Journal of Mechanical Design* 131(3): 031001(13).

Osorio-Urzúa, C.A. (2007). “Architectural Innovations, Functional Emergence and Diversification in Engineering Systems.” *Technology, Management and Policy*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Padula, S.L., Sandridge, S.A., Haftka, R.T., and Walsh, J.L. (1989) “Demonstration of Decomposition and Optimization in the Design of Experimental Space Systems.” *NASA Technical Reports Server* 19890015789.

Parkin, K.L.G., Sercel, J.C., Liu, M.J., and Thunnissen, D.P. (2003). “ICEMaker^{TM}: An Excel-Based Environment for Collaborative Design.” *Proceedings of the IEEE Aerospace Conference*. IEEE, Big Sky, MT, Vol. 8, pp. 8_3669- 8_3679.

Pawar, K.S. (1994). “Organizational and Managerial Issues.” In *Concurrent Engineering: Concepts, Implementation and Practice*, eds. C.S. Syan and U. Menon, pp. 49-74. London, UK: Chapman & Hall.

Phillips, K.W. and Loyd, D.L. (2006). “When Surface and Deep-Level Diversity Collide: The Effects on Dissenting Group Members.” *Organizational Behavior and Human Decision Processes* 99(2): 143-160.

Prasad, B. (1996). *Concurrent Engineering Fundamentals, Volume I: Integrated Product and Process Organization*. Upper Saddle River, NJ: Prentice Hall PTR.

Reeves, E.I. (1999). “Spacecraft Design and Sizing.” In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 301-352. El Segundo, CA: Microcosm.

Robinson, G.L. (2008). “Systems Engineering Initiatives at NASA.” *Goddard/SMA-D Education Series*, 25 Sept 2008.

Rogers, J.L. (1999). “Tools and Techniques for Decomposing and Managing Complex Design Projects.” *Journal of Aircraft* 36(1): 266-274.

Ross, A.M. (2006). “Managing Unarticulated Value: Changeability in Multi-Attribute Tradespace Exploration.” *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Rouse, W.B., Cannon-Bowers, J.A., and Salas, E. (1992). “The Role of Mental Models in Team Performance in Complex Systems.” *IEEE Transactions on Systems, Man, and Cybernetics* 22(6): 1296-1308.

Rouse, W.B. and Morris, N.M. (1986). “On Looking Into the Black Box: Prospects and Limits in the Search for Mental Models.” *Psychological Bulletin* 100(3): 349-363.

Sage, A.P. (1992). *Systems Engineering*. New York, NY: John Wiley & Sons.

Salas, E., Kosarzycki, M.P., Tannenbaum, S.I., and Carnegie, D. (2004). "Principles and Advice for Understanding and Promoting Effective Teamwork in Organizations." In *Leading in Turbulent Times: Managing in the New World of Work*, eds. R.J. Burke and C.L. Cooper, pp. 95-120. Malden, MA: Blackwell.

Salas, E., Rosen, M.A., Burke, C.S, Goodwin, G.F., and Fiore, S.M. (2006). "The Making of a Dream Team: When Expert Teams Do Best." In *The Cambridge Handbook of Expertise and Expert Performance*, eds. K.A. Ericsson, N. Charness, R.R. Hoffman, and P.J. Feltovich, pp. 439-454. Cambridge, UK: Cambridge University Press.

Sanders, G.L. (2002). "Cool as ICE: The Sky's No Limit for BSS Design Center." *Boeing Frontiers Online* 1(4). Update Aug 2002. View 10 Mar 2009. <http://www.boeing.com/news/frontiers/archive/2002/august/i_ids2.html>.

Santarek, K. and Buseif, I.M. (1998) "Modelling and Design of Flexible Manufacturing Systems Using SADT and Petri Nets Tools." *Journal of Materials Processing Technology* 76(1): 212-218.

Schein, E.H. (1996). "Three Cultures of Management: The Key to Organizational Learning." *Sloan Management Review* 38(1): 9-20.

Sercel, J., Sepahban, S., and Wall, S. (1998). "ICE Heats Up Design Productivity." *Aerospace America*, Jul 1998: 20-22.

Smaling, R.M. (2005). "System Architecture Analysis and Selection Under Uncertainty." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Smith, J.L. (1998). "Concurrent Engineering in the Jet Propulsion Laboratory Project Design Center." *Aerospace Manufacturing Technology Conference & Exposition*. Society of Automotive Engineers, Long Beach, CA.

Sosa, M.E., Eppinger, S.D., and Rowles, C.M. (2003). "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions." *Journal of Mechanical Design* 125(2): 240-252.

Stagney, D.B. (2003). "The Integrated Concurrent Enterprise." *Aeronautics and Astronautics*. Massachusetts Institute of Technology, Cambridge, MA. **S.M.**

Stanke, A.K. (2006). "Creating High Performance Enterprises." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Stasser, G., Stewart, D.D., and Wittenbaum, G.M. (1995). "Expert Roles and Information Exchange During Discussion: The Importance of Knowing Who Knows What." *Journal of Experimental Social Psychology* 31(3): 244-265.

Steward, D.V. (1981a). "The Design Structure System: A Method for Managing the Design of Complex Systems." *IEEE Transactions on Engineering Management* 28: 71-74.

Steward, D.V. (1981b). *Systems Analysis and Management: Structure, Strategy and Design*. New York, NY: Petrocelli.

Stout, R.J., Cannon-Bowers, J.A., Salas, E., and Milanovich, D.M. (1999). "Planning, Shared Mental Models, and Coordinated Performance: An Empirical Link Is Established." *Human Factors* 41(1): 61-71.

Suh, E.S. (2005). "Flexible Product Platforms." *Engineering Systems*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Tang, V. (2006). "Corporate Decision Analysis: An Engineering Approach." *Interdisciplinary Studies*. Massachusetts Institute of Technology, Cambridge, MA. **Ph.D.**

Temponi, C., Yen, J., and Tiao, W.A. (1999) "House of Quality: A Fuzzy Logic-Based Requirements Analysis." *European Journal of Operational Research* 117(2): 340-354.

Ulrich, K.T. and Eppinger, S.D. (2004). *Product Design and Development*. Boston, MA: McGraw-Hill/Irwin.

Van Eikema Hommes, Q.D. (2008). "Comparison and Application of Metrics that Define the Components Modularity in Complex Products." *Proceedings of the International Design Engineering Technical Conferences & Computers and Information in Engineering Conference,* American Society of Mechanical Engineers, Brooklyn, NY, pp. 1-10.

Wall, S.D. (1999). "Reinventing the Design Process: Teams and Models." *Specialist Symposium on Novel Concepts for Smaller, Faster and Better Space Missions*. International Astronautical Federation, Redondo Beach, CA.

Wasserman, S. and Faust, K. (1999). *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge, UK: Cambridge University Press.

Wertz, J.R. (1999). "Space Mission Geometry." In *Space Mission Analysis and Design*, 3rd ed., eds. J.R. Wertz and W.J. Larson, pp. 95-130. El Segundo, CA: Microcosm.

Wertz, J.R. and Larson, W.J. (1999). *Space Mission Analysis and Design*, 3rd ed. El Segundo, CA: Microcosm.

Womack, J.,P., Jones, D.T., Roos, D., and Carpenter, D.S. (1990). *The Machine That Changed the World: The Story of Lean Production*. New York, NY: Rawson.

Woods, D.D. and Hollnagel, E. (2006). *Joint Cognitive Systems: Patterns in Cognitive Systems Engineering*. Boca Raton, FL: Taylor and Francis.

Yang, M.C. and Ji, H. (2007). "A Text-Based Analysis Approach to Representing the Design Selection Process." *International Conference on Engineering Design*. The Design Society, Paris, France, pp. 1-10.

Appendix A

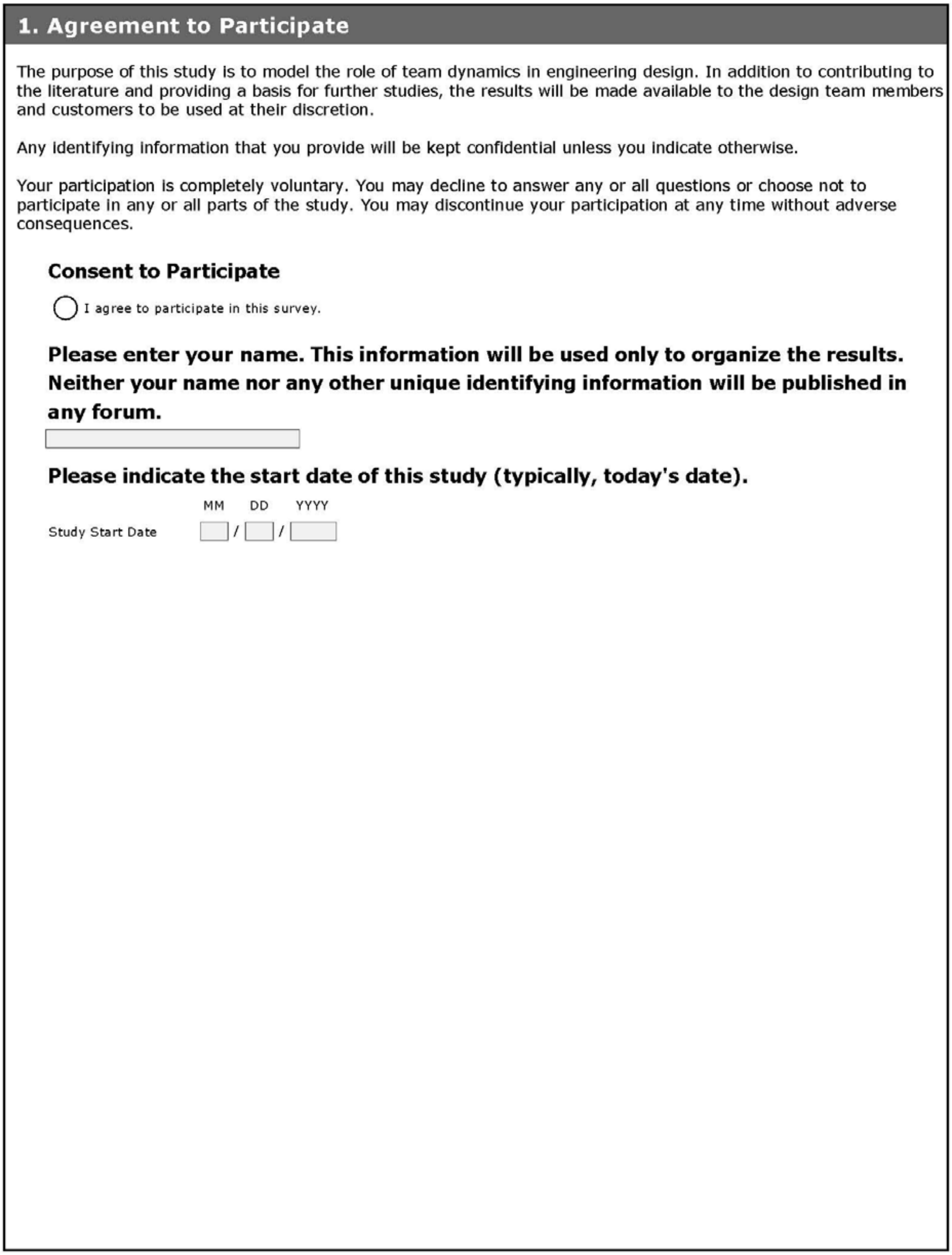
Sample Design Session Surveys

This appendix provides a sample of each of the surveys distributed to the design team and the customer team before and after each session of the Mission Design Laboratory (MDL). It is divided into two sections. The first section shows the pre-session survey that was distributed before each session, and the second section shows the post-session survey that was distributed after each session.

The question on major design drivers is repeated exactly as-is on the pre-session and post-session surveys so that change in shared knowledge over time could be measured. The remaining substantive survey questions are on the post-session survey because their purpose is to collect data on the events, content, and outcomes of the design sessions. The question on interactions with other team members was used to construct the reported interaction matrix for each design session, and the questions on Technology Readiness Level (TRL) were used to collect data on the maturity of each subsystem's technology and to compute the system-level metric for mission concept maturity, Mission TRL (MTRL). Finally, the long survey question on parameter flow in the design session represents the second step of the procedure for constructing the Design Structure Matrix (DSM) described in Chapter 4.

For the purposes of the model of shared knowledge in the team, no distinction is made in these surveys between design team and customer team members. The difference, however, could be important to future work. Thus, the separate customer team roles are recorded in the survey and should be made explicit to the extent that it is relevant when reproducing or extending the work presented in the thesis.

A.1. Sample Pre-Session Survey



1. Agreement to Participate

The purpose of this study is to model the role of team dynamics in engineering design. In addition to contributing to the literature and providing a basis for further studies, the results will be made available to the design team members and customers to be used at their discretion.

Any identifying information that you provide will be kept confidential unless you indicate otherwise.

Your participation is completely voluntary. You may decline to answer any or all questions or choose not to participate in any or all parts of the study. You may discontinue your participation at any time without adverse consequences.

Consent to Participate

○ I agree to participate in this survey.

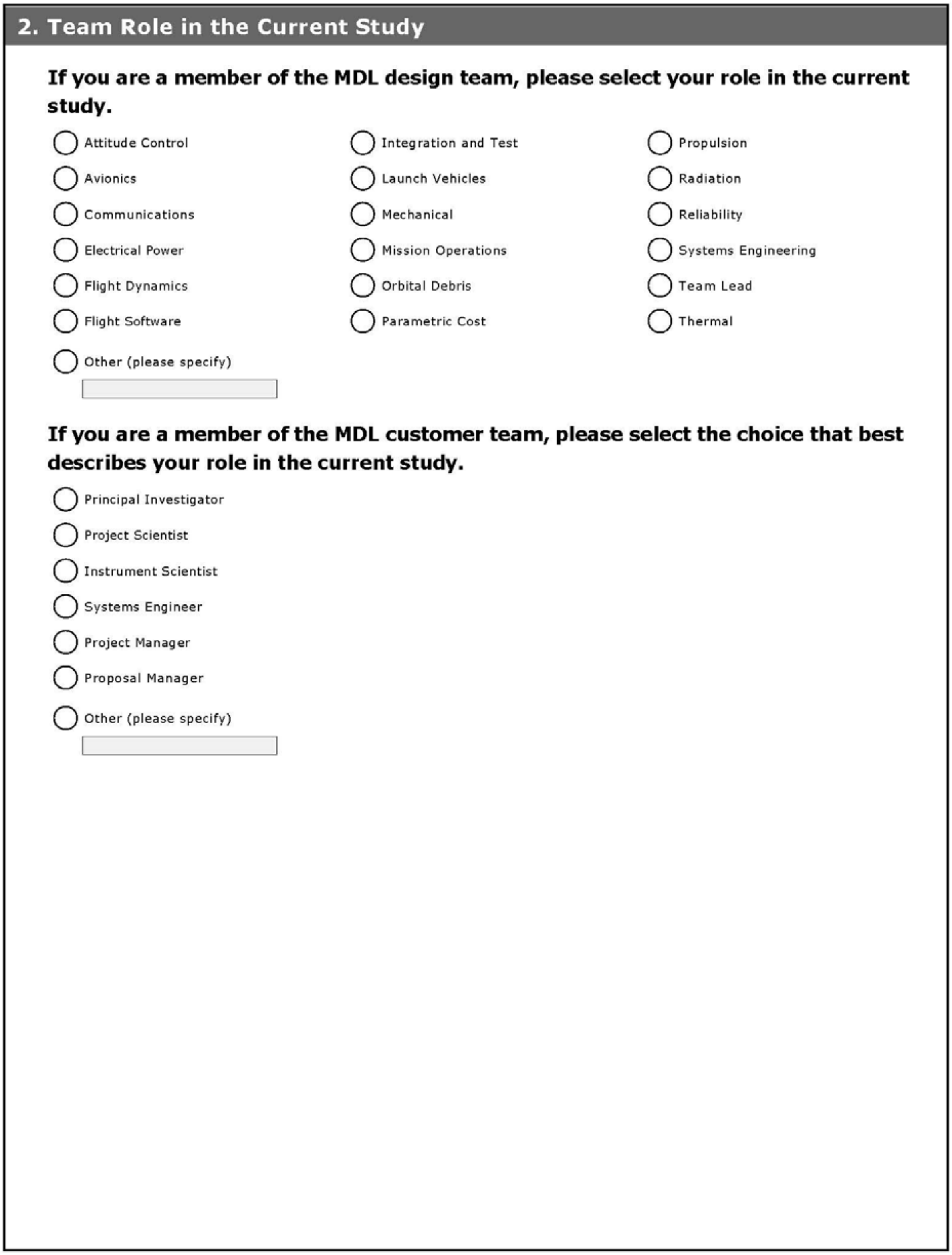
Please enter your name. This information will be used only to organize the results. Neither your name nor any other unique identifying information will be published in any forum.

☐

Please indicate the start date of this study (typically, today's date).

Study Start Date ☐ MM / ☐ DD / ☐ YYYY

Sample Pre-Session Survey (Cont.)



2. Team Role in the Current Study

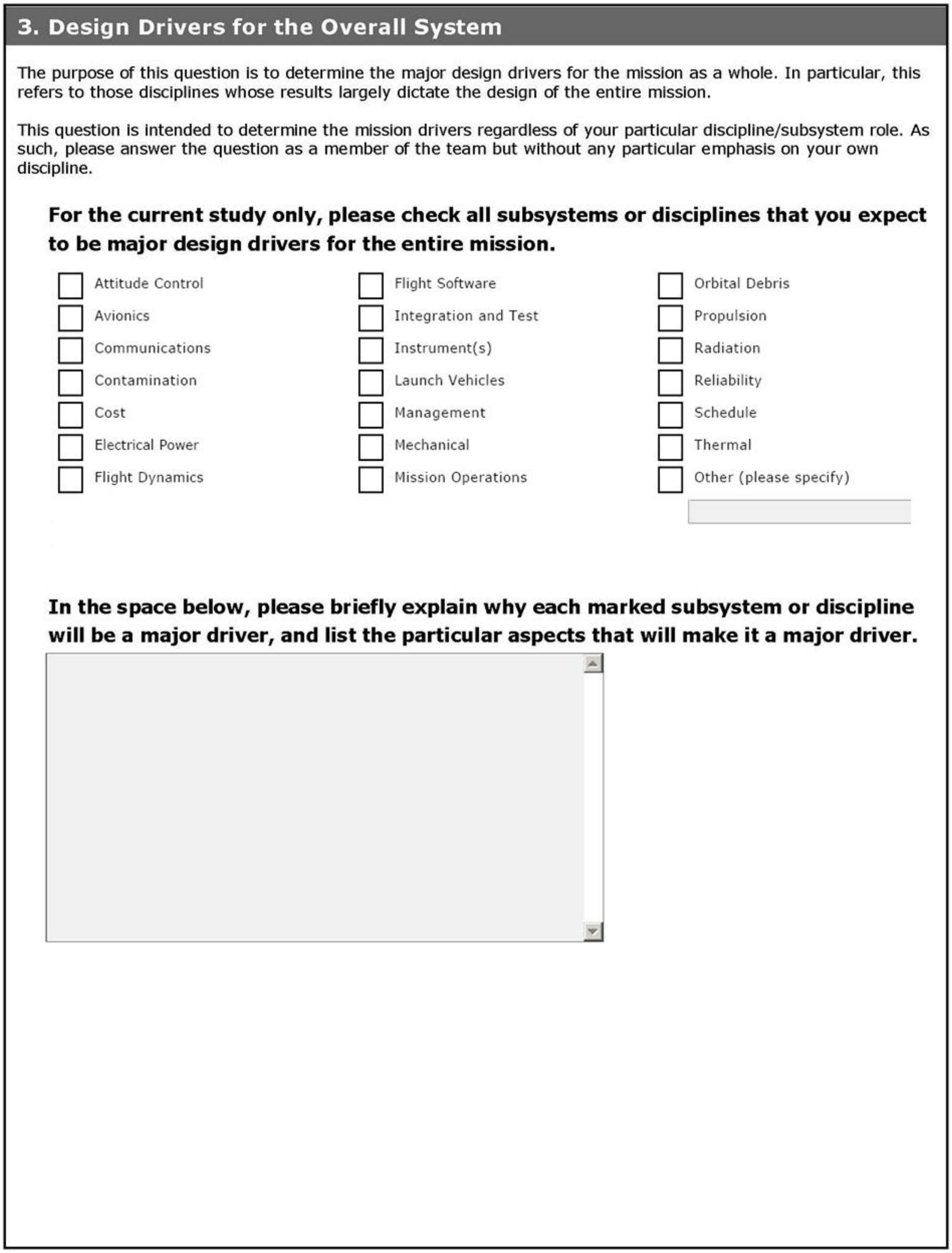
If you are a member of the MDL design team, please select your role in the current study.

○ Attitude Control
○ Avionics
○ Communications
○ Electrical Power
○ Flight Dynamics
○ Flight Software
○ Integration and Test
○ Launch Vehicles
○ Mechanical
○ Mission Operations
○ Orbital Debris
○ Parametric Cost
○ Propulsion
○ Radiation
○ Reliability
○ Systems Engineering
○ Team Lead
○ Thermal
○ Other (please specify)

If you are a member of the MDL customer team, please select the choice that best describes your role in the current study.

○ Principal Investigator
○ Project Scientist
○ Instrument Scientist
○ Systems Engineer
○ Project Manager
○ Proposal Manager
○ Other (please specify)

Sample Pre-Session Survey (Cont.)



3. Design Drivers for the Overall System

The purpose of this question is to determine the major design drivers for the mission as a whole. In particular, this refers to those disciplines whose results largely dictate the design of the entire mission.

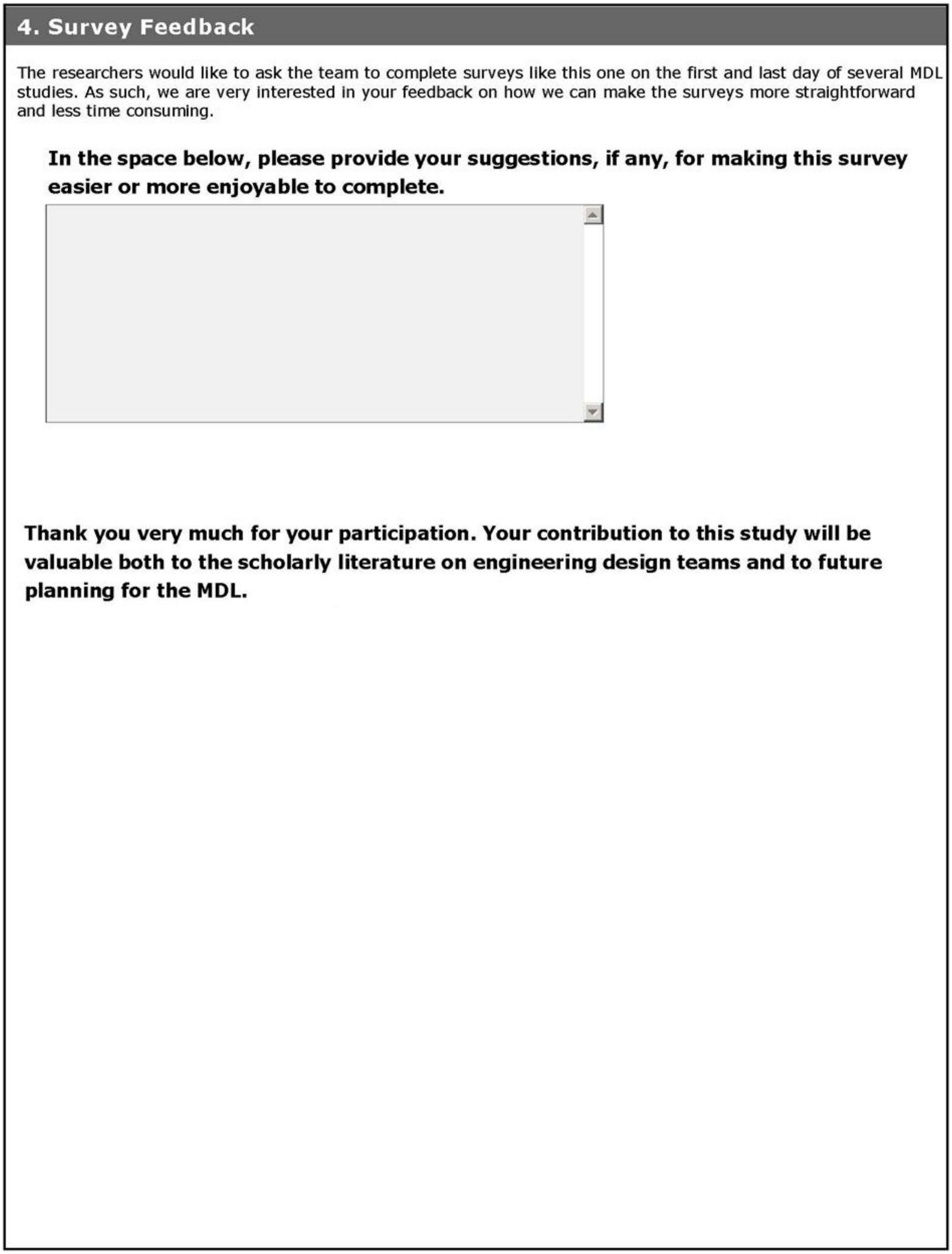
This question is intended to determine the mission drivers regardless of your particular discipline/subsystem role. As such, please answer the question as a member of the team but without any particular emphasis on your own discipline.

For the current study only, please check all subsystems or disciplines that you expect to be major design drivers for the entire mission.

☐ Attitude Control
☐ Avionics
☐ Communications
☐ Contamination
☐ Cost
☐ Electrical Power
☐ Flight Dynamics
☐ Flight Software
☐ Integration and Test
☐ Instrument(s)
☐ Launch Vehicles
☐ Management
☐ Mechanical
☐ Mission Operations
☐ Orbital Debris
☐ Propulsion
☐ Radiation
☐ Reliability
☐ Schedule
☐ Thermal
☐ Other (please specify)

In the space below, please briefly explain why each marked subsystem or discipline will be a major driver, and list the particular aspects that will make it a major driver.

Sample Pre-Session Survey (Cont.)



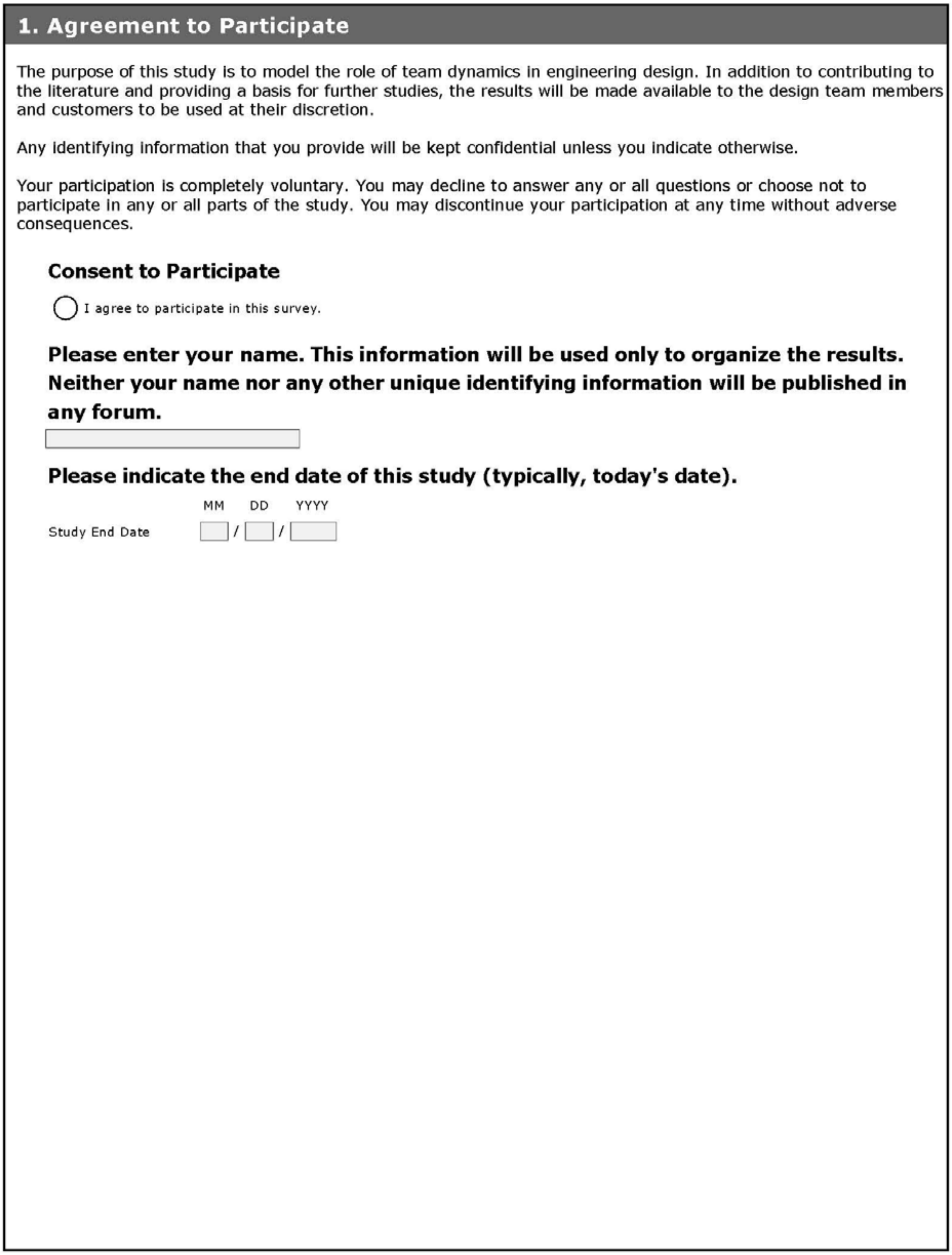
4. Survey Feedback

The researchers would like to ask the team to complete surveys like this one on the first and last day of several MDL studies. As such, we are very interested in your feedback on how we can make the surveys more straightforward and less time consuming.

In the space below, please provide your suggestions, if any, for making this survey easier or more enjoyable to complete.

Thank you very much for your participation. Your contribution to this study will be valuable both to the scholarly literature on engineering design teams and to future planning for the MDL.

A.2. Sample Post-Session Survey



1. Agreement to Participate

The purpose of this study is to model the role of team dynamics in engineering design. In addition to contributing to the literature and providing a basis for further studies, the results will be made available to the design team members and customers to be used at their discretion.

Any identifying information that you provide will be kept confidential unless you indicate otherwise.

Your participation is completely voluntary. You may decline to answer any or all questions or choose not to participate in any or all parts of the study. You may discontinue your participation at any time without adverse consequences.

Consent to Participate

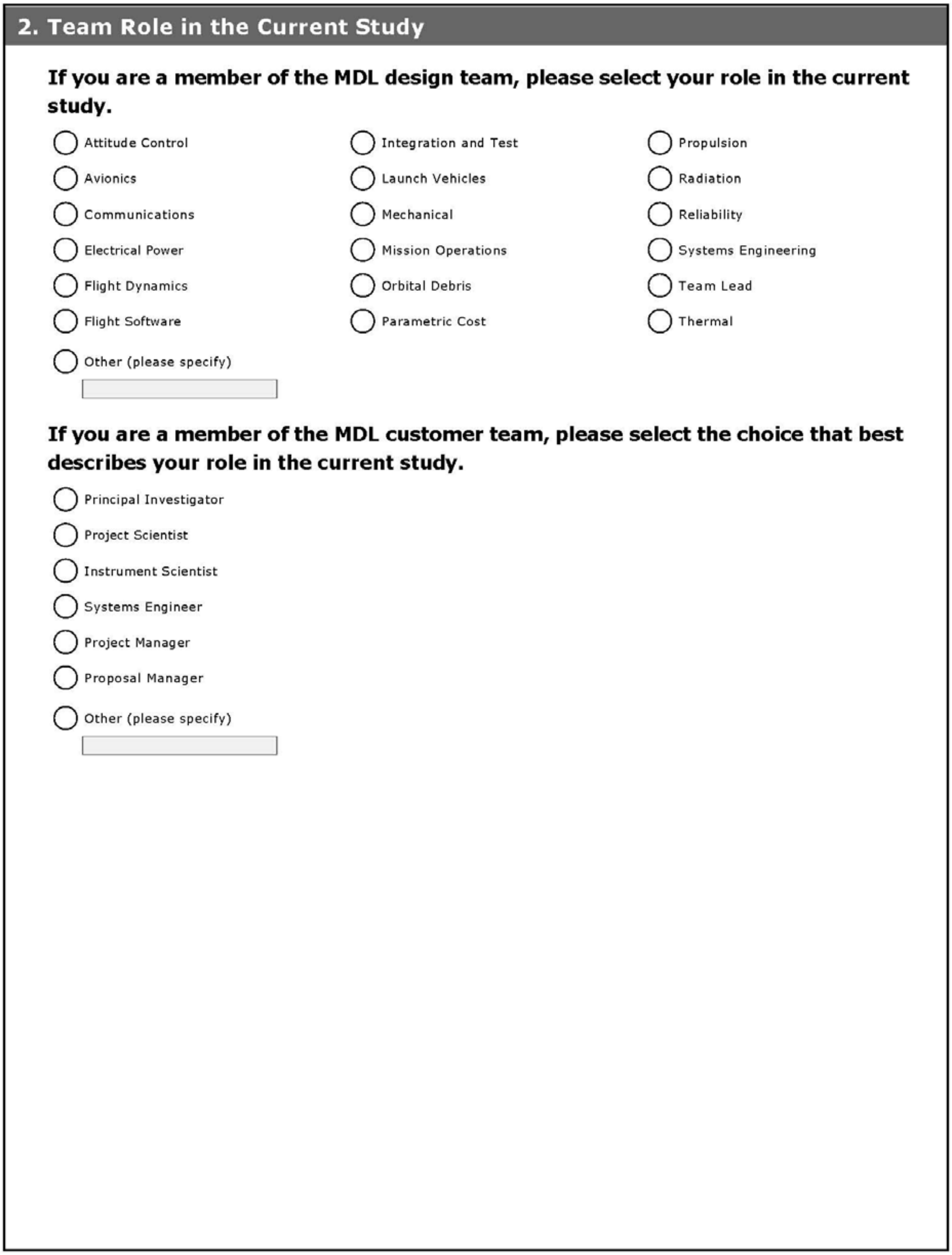
○ I agree to participate in this survey.

Please enter your name. This information will be used only to organize the results. Neither your name nor any other unique identifying information will be published in any forum.

Please indicate the end date of this study (typically, today's date).

Study End Date MM / DD / YYYY

Sample Post-Session Survey (Cont.)



2. Team Role in the Current Study

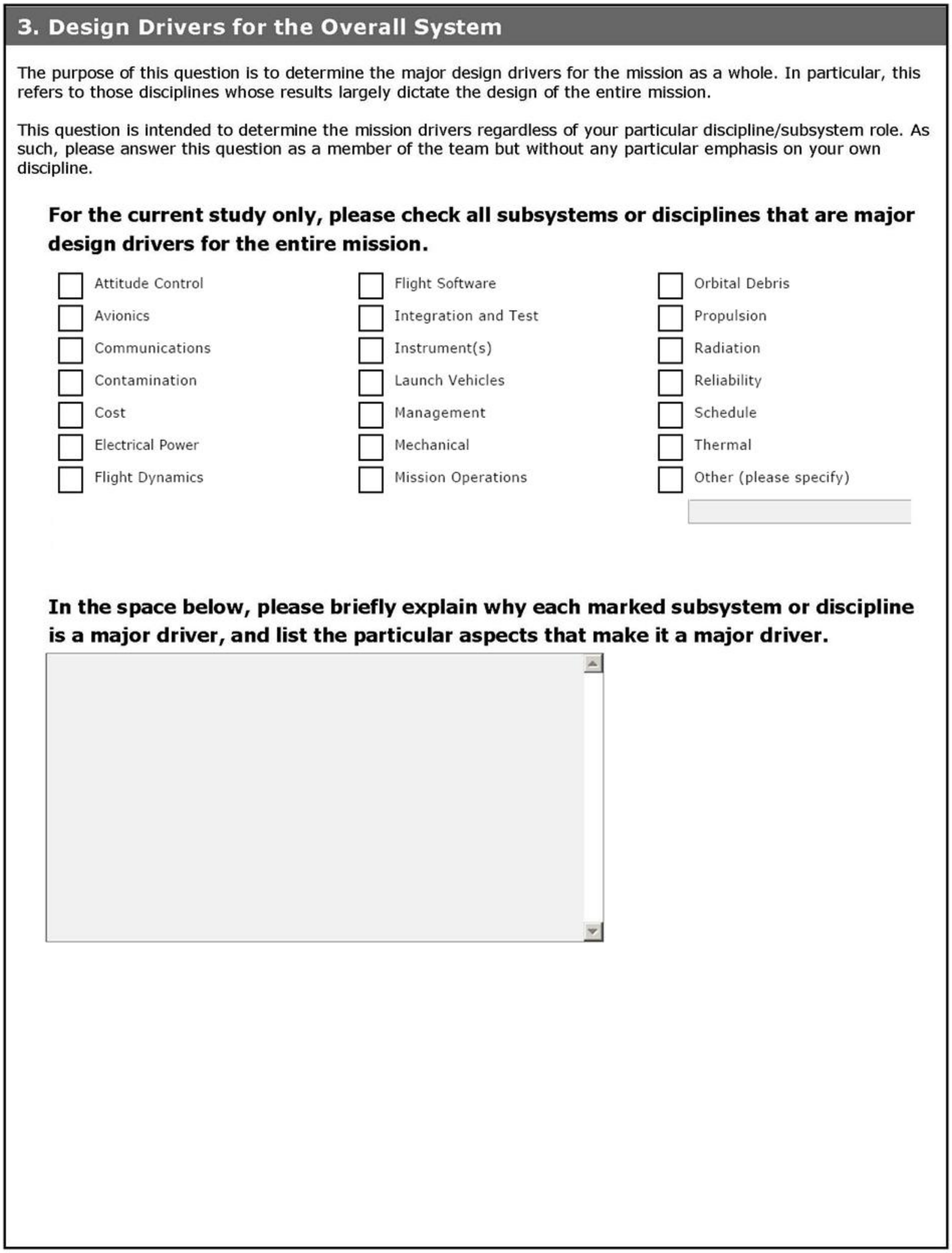
If you are a member of the MDL design team, please select your role in the current study.

○ Attitude Control
○ Avionics
○ Communications
○ Electrical Power
○ Flight Dynamics
○ Flight Software
○ Integration and Test
○ Launch Vehicles
○ Mechanical
○ Mission Operations
○ Orbital Debris
○ Parametric Cost
○ Propulsion
○ Radiation
○ Reliability
○ Systems Engineering
○ Team Lead
○ Thermal
○ Other (please specify)

If you are a member of the MDL customer team, please select the choice that best describes your role in the current study.

○ Principal Investigator
○ Project Scientist
○ Instrument Scientist
○ Systems Engineer
○ Project Manager
○ Proposal Manager
○ Other (please specify)

Sample Post-Session Survey (Cont.)



3. Design Drivers for the Overall System

The purpose of this question is to determine the major design drivers for the mission as a whole. In particular, this refers to those disciplines whose results largely dictate the design of the entire mission.

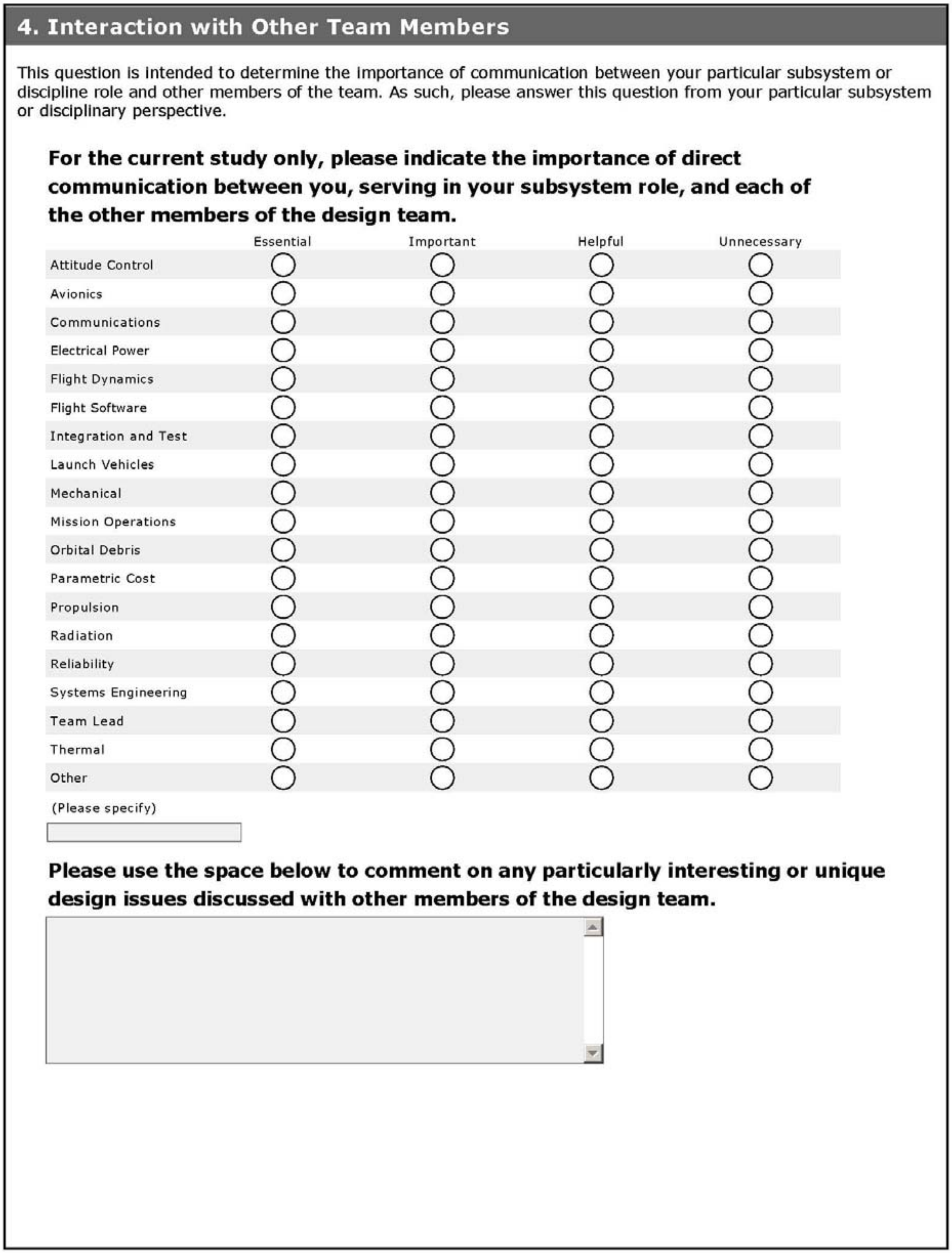
This question is intended to determine the mission drivers regardless of your particular discipline/subsystem role. As such, please answer this question as a member of the team but without any particular emphasis on your own discipline.

For the current study only, please check all subsystems or disciplines that are major design drivers for the entire mission.

☐ Attitude Control
☐ Avionics
☐ Communications
☐ Contamination
☐ Cost
☐ Electrical Power
☐ Flight Dynamics
☐ Flight Software
☐ Integration and Test
☐ Instrument(s)
☐ Launch Vehicles
☐ Management
☐ Mechanical
☐ Mission Operations
☐ Orbital Debris
☐ Propulsion
☐ Radiation
☐ Reliability
☐ Schedule
☐ Thermal
☐ Other (please specify)

In the space below, please briefly explain why each marked subsystem or discipline is a major driver, and list the particular aspects that make it a major driver.

Sample Post-Session Survey (Cont.)



4. Interaction with Other Team Members

This question is intended to determine the importance of communication between your particular subsystem or discipline role and other members of the team. As such, please answer this question from your particular subsystem or disciplinary perspective.

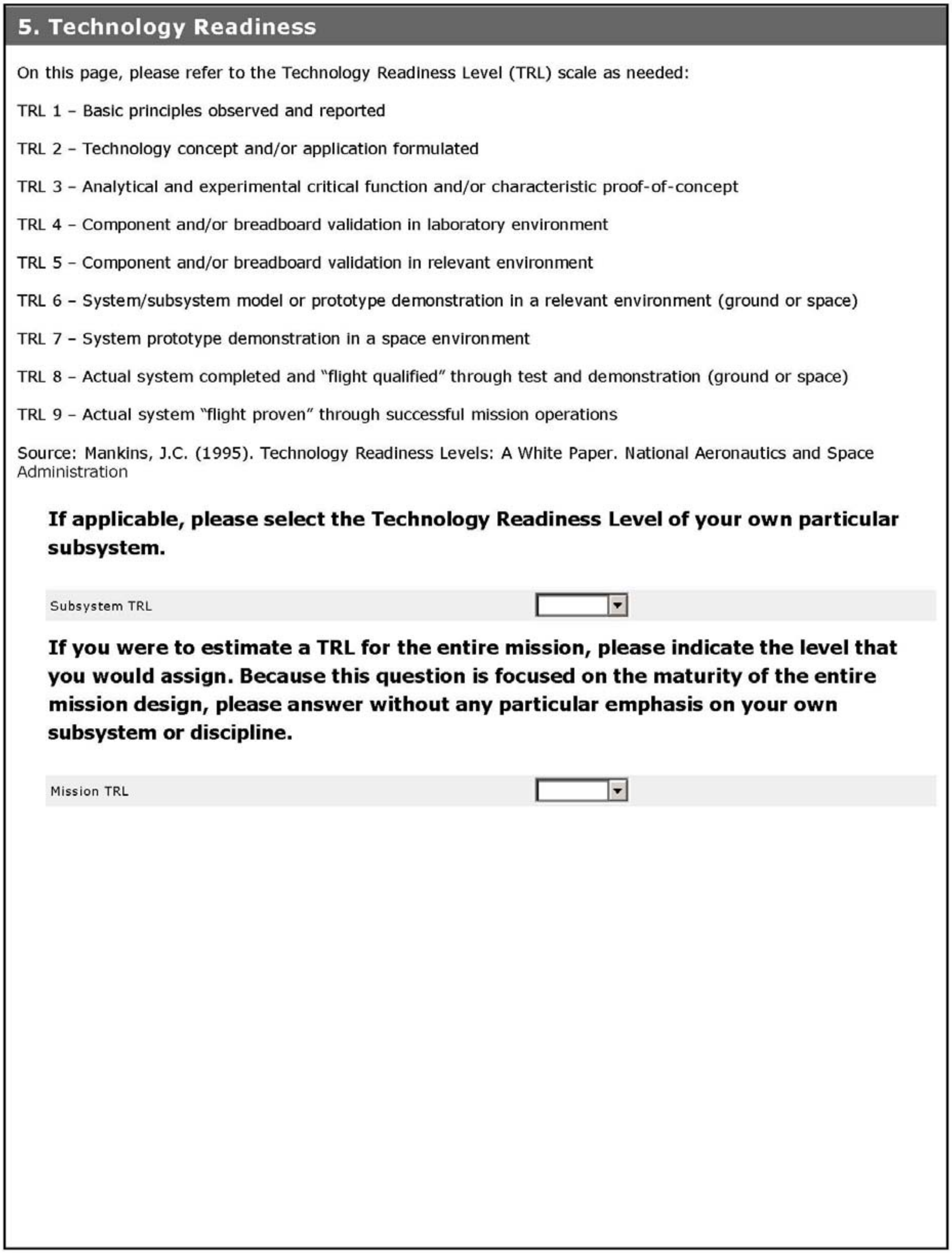
For the current study only, please indicate the importance of direct communication between you, serving in your subsystem role, and each of the other members of the design team.

	Essential	Important	Helpful	Unnecessary
Attitude Control	○	○	○	○
Avionics	○	○	○	○
Communications	○	○	○	○
Electrical Power	○	○	○	○
Flight Dynamics	○	○	○	○
Flight Software	○	○	○	○
Integration and Test	○	○	○	○
Launch Vehicles	○	○	○	○
Mechanical	○	○	○	○
Mission Operations	○	○	○	○
Orbital Debris	○	○	○	○
Parametric Cost	○	○	○	○
Propulsion	○	○	○	○
Radiation	○	○	○	○
Reliability	○	○	○	○
Systems Engineering	○	○	○	○
Team Lead	○	○	○	○
Thermal	○	○	○	○
Other	○	○	○	○

(Please specify)

Please use the space below to comment on any particularly interesting or unique design issues discussed with other members of the design team.

Sample Post-Session Survey (Cont.)



5. Technology Readiness

On this page, please refer to the Technology Readiness Level (TRL) scale as needed:

TRL 1 – Basic principles observed and reported

TRL 2 – Technology concept and/or application formulated

TRL 3 – Analytical and experimental critical function and/or characteristic proof-of-concept

TRL 4 – Component and/or breadboard validation in laboratory environment

TRL 5 – Component and/or breadboard validation in relevant environment

TRL 6 – System/subsystem model or prototype demonstration in a relevant environment (ground or space)

TRL 7 – System prototype demonstration in a space environment

TRL 8 – Actual system completed and "flight qualified" through test and demonstration (ground or space)

TRL 9 – Actual system "flight proven" through successful mission operations

Source: Mankins, J.C. (1995). Technology Readiness Levels: A White Paper. National Aeronautics and Space Administration

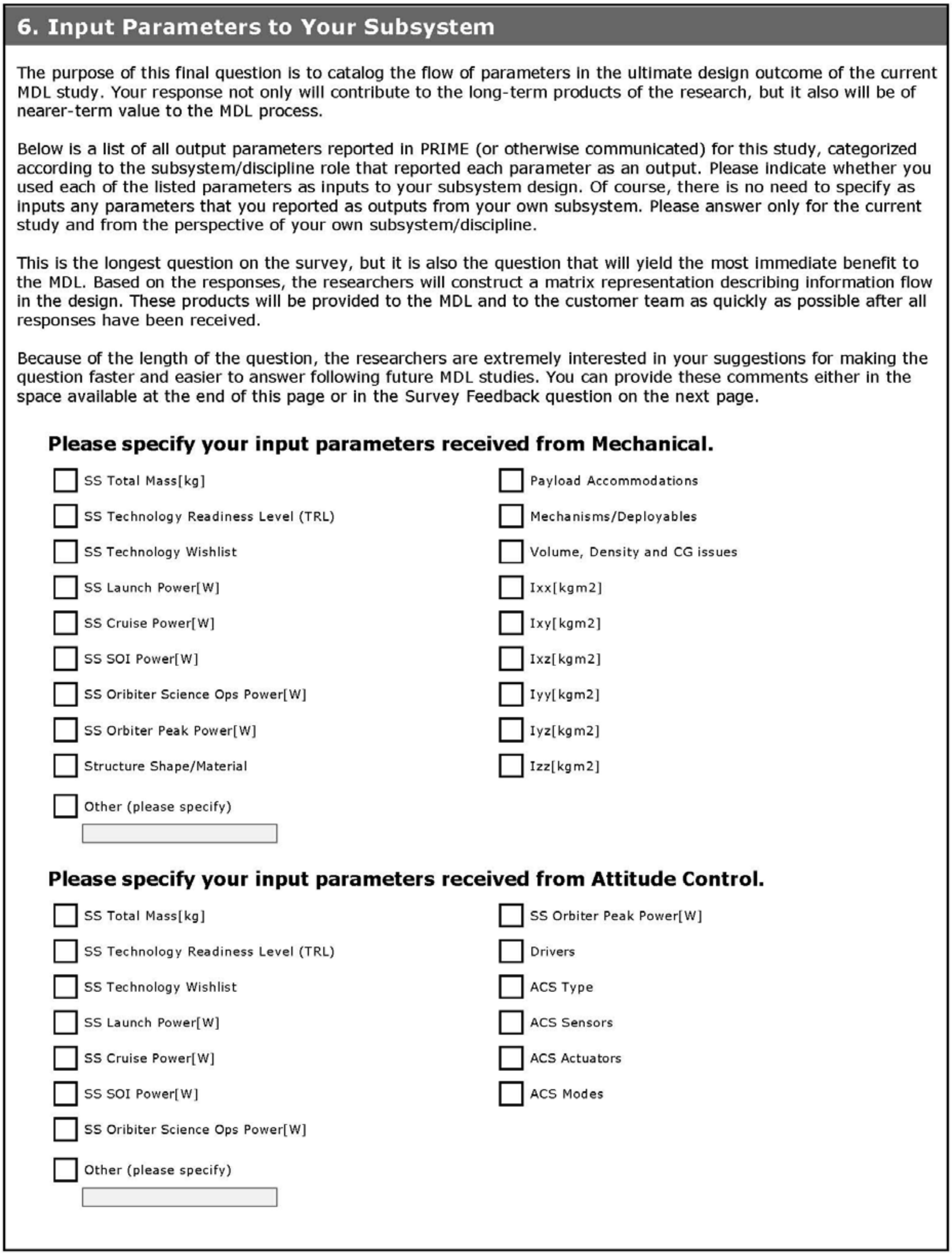
If applicable, please select the Technology Readiness Level of your own particular subsystem.

Subsystem TRL

If you were to estimate a TRL for the entire mission, please indicate the level that you would assign. Because this question is focused on the maturity of the entire mission design, please answer without any particular emphasis on your own subsystem or discipline.

Mission TRL

Sample Post-Session Survey (Cont.)



6. Input Parameters to Your Subsystem

The purpose of this final question is to catalog the flow of parameters in the ultimate design outcome of the current MDL study. Your response not only will contribute to the long-term products of the research, but it also will be of nearer-term value to the MDL process.

Below is a list of all output parameters reported in PRIME (or otherwise communicated) for this study, categorized according to the subsystem/discipline role that reported each parameter as an output. Please indicate whether you used each of the listed parameters as inputs to your subsystem design. Of course, there is no need to specify as inputs any parameters that you reported as outputs from your own subsystem. Please answer only for the current study and from the perspective of your own subsystem/discipline.

This is the longest question on the survey, but it is also the question that will yield the most immediate benefit to the MDL. Based on the responses, the researchers will construct a matrix representation describing information flow in the design. These products will be provided to the MDL and to the customer team as quickly as possible after all responses have been received.

Because of the length of the question, the researchers are extremely interested in your suggestions for making the question faster and easier to answer following future MDL studies. You can provide these comments either in the space available at the end of this page or in the Survey Feedback question on the next page.

Please specify your input parameters received from Mechanical.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ SS Orbiter Peak Power[W]
☐ Structure Shape/Material
☐ Payload Accommodations
☐ Mechanisms/Deployables
☐ Volume, Density and CG issues
☐ Ixx[kgm2]
☐ Ixy[kgm2]
☐ Ixz[kgm2]
☐ Iyy[kgm2]
☐ Iyz[kgm2]
☐ Izz[kgm2]

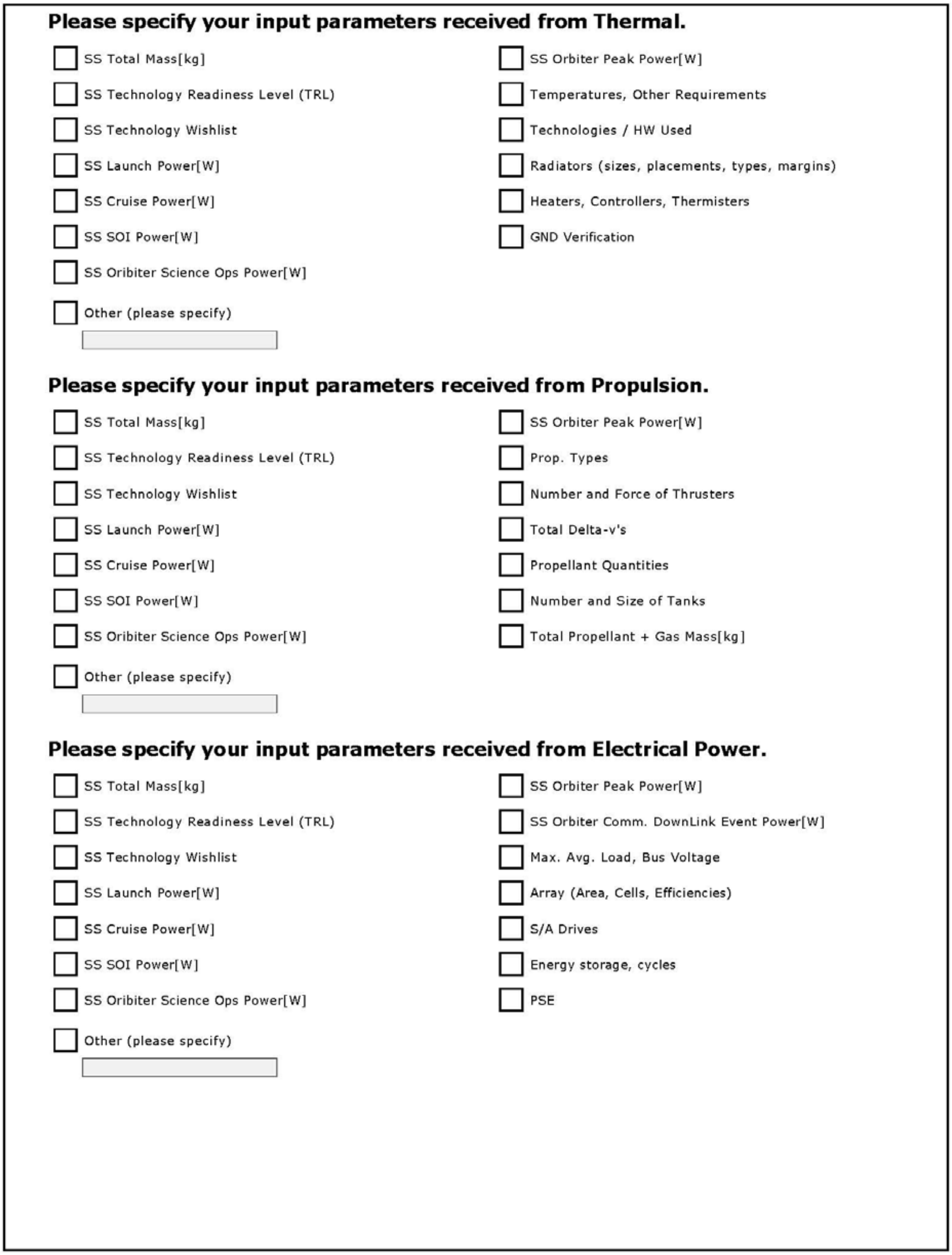
☐ Other (please specify)

Please specify your input parameters received from Attitude Control.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ SS Orbiter Peak Power[W]
☐ Drivers
☐ ACS Type
☐ ACS Sensors
☐ ACS Actuators
☐ ACS Modes

☐ Other (please specify)

Sample Post-Session Survey (Cont.)



Please specify your input parameters received from Thermal.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ Other (please specify)

☐ SS Orbiter Peak Power[W]
☐ Temperatures, Other Requirements
☐ Technologies / HW Used
☐ Radiators (sizes, placements, types, margins)
☐ Heaters, Controllers, Thermisters
☐ GND Verification

Please specify your input parameters received from Propulsion.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ Other (please specify)

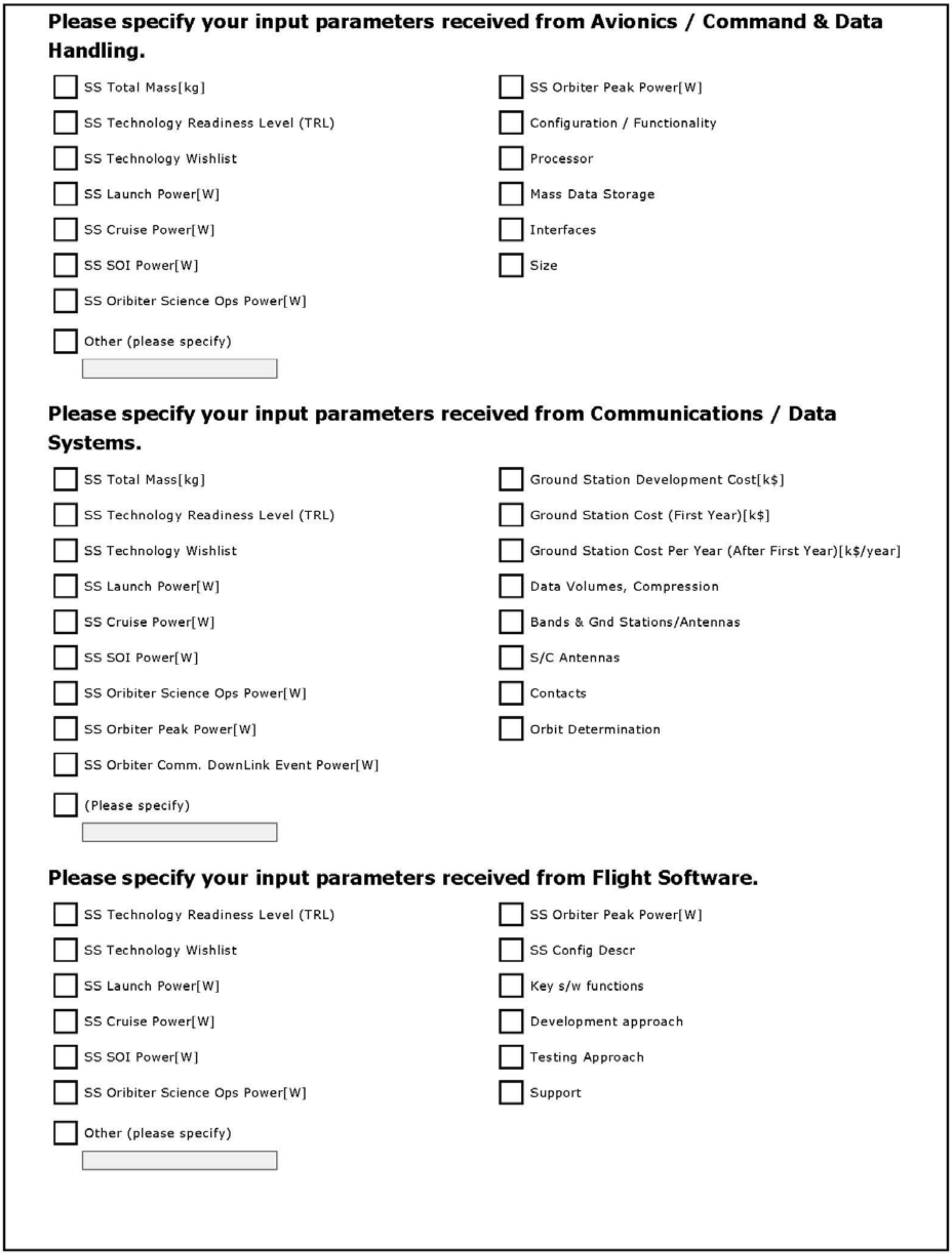
☐ SS Orbiter Peak Power[W]
☐ Prop. Types
☐ Number and Force of Thrusters
☐ Total Delta-v's
☐ Propellant Quantities
☐ Number and Size of Tanks
☐ Total Propellant + Gas Mass[kg]

Please specify your input parameters received from Electrical Power.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ Other (please specify)

☐ SS Orbiter Peak Power[W]
☐ SS Orbiter Comm. DownLink Event Power[W]
☐ Max. Avg. Load, Bus Voltage
☐ Array (Area, Cells, Efficiencies)
☐ S/A Drives
☐ Energy storage, cycles
☐ PSE

Sample Post-Session Survey (Cont.)



Please specify your input parameters received from Avionics / Command & Data Handling.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ Other (please specify)

☐ SS Orbiter Peak Power[W]
☐ Configuration / Functionality
☐ Processor
☐ Mass Data Storage
☐ Interfaces
☐ Size

Please specify your input parameters received from Communications / Data Systems.

☐ SS Total Mass[kg]
☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ SS Orbiter Peak Power[W]
☐ SS Orbiter Comm. DownLink Event Power[W]
☐ (Please specify)

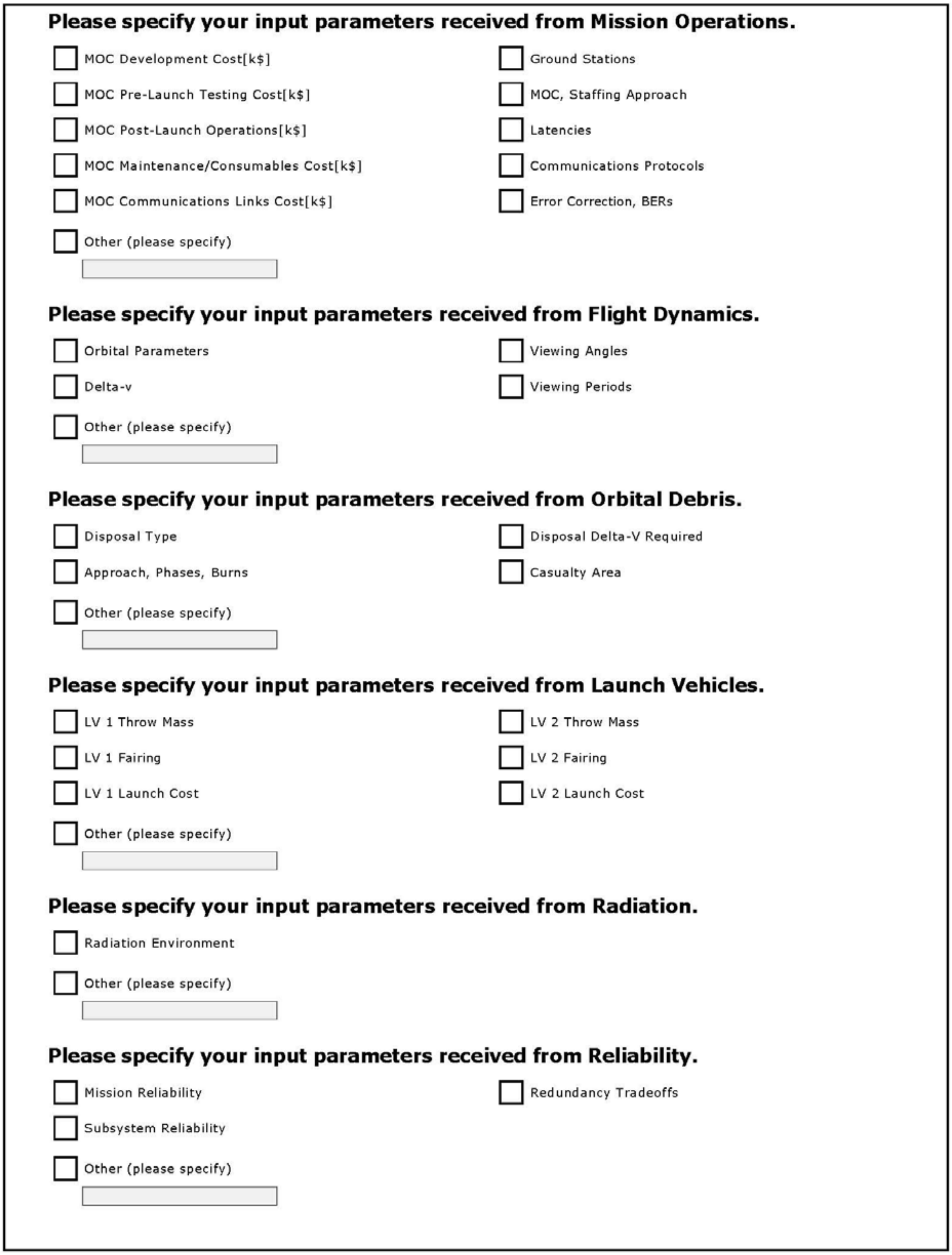
☐ Ground Station Development Cost[k$]
☐ Ground Station Cost (First Year)[k$]
☐ Ground Station Cost Per Year (After First Year)[k$/year]
☐ Data Volumes, Compression
☐ Bands & Gnd Stations/Antennas
☐ S/C Antennas
☐ Contacts
☐ Orbit Determination

Please specify your input parameters received from Flight Software.

☐ SS Technology Readiness Level (TRL)
☐ SS Technology Wishlist
☐ SS Launch Power[W]
☐ SS Cruise Power[W]
☐ SS SOI Power[W]
☐ SS Oribiter Science Ops Power[W]
☐ Other (please specify)

☐ SS Orbiter Peak Power[W]
☐ SS Config Descr
☐ Key s/w functions
☐ Development approach
☐ Testing Approach
☐ Support

Sample Post-Session Survey (Cont.)



Please specify your input parameters received from Mission Operations.

☐ MOC Development Cost[k$]
☐ MOC Pre-Launch Testing Cost[k$]
☐ MOC Post-Launch Operations[k$]
☐ MOC Maintenance/Consumables Cost[k$]
☐ MOC Communications Links Cost[k$]
☐ Other (please specify)
☐ Ground Stations
☐ MOC, Staffing Approach
☐ Latencies
☐ Communications Protocols
☐ Error Correction, BERs

Please specify your input parameters received from Flight Dynamics.

☐ Orbital Parameters
☐ Delta-v
☐ Other (please specify)
☐ Viewing Angles
☐ Viewing Periods

Please specify your input parameters received from Orbital Debris.

☐ Disposal Type
☐ Approach, Phases, Burns
☐ Other (please specify)
☐ Disposal Delta-V Required
☐ Casualty Area

Please specify your input parameters received from Launch Vehicles.

☐ LV 1 Throw Mass
☐ LV 1 Fairing
☐ LV 1 Launch Cost
☐ Other (please specify)
☐ LV 2 Throw Mass
☐ LV 2 Fairing
☐ LV 2 Launch Cost

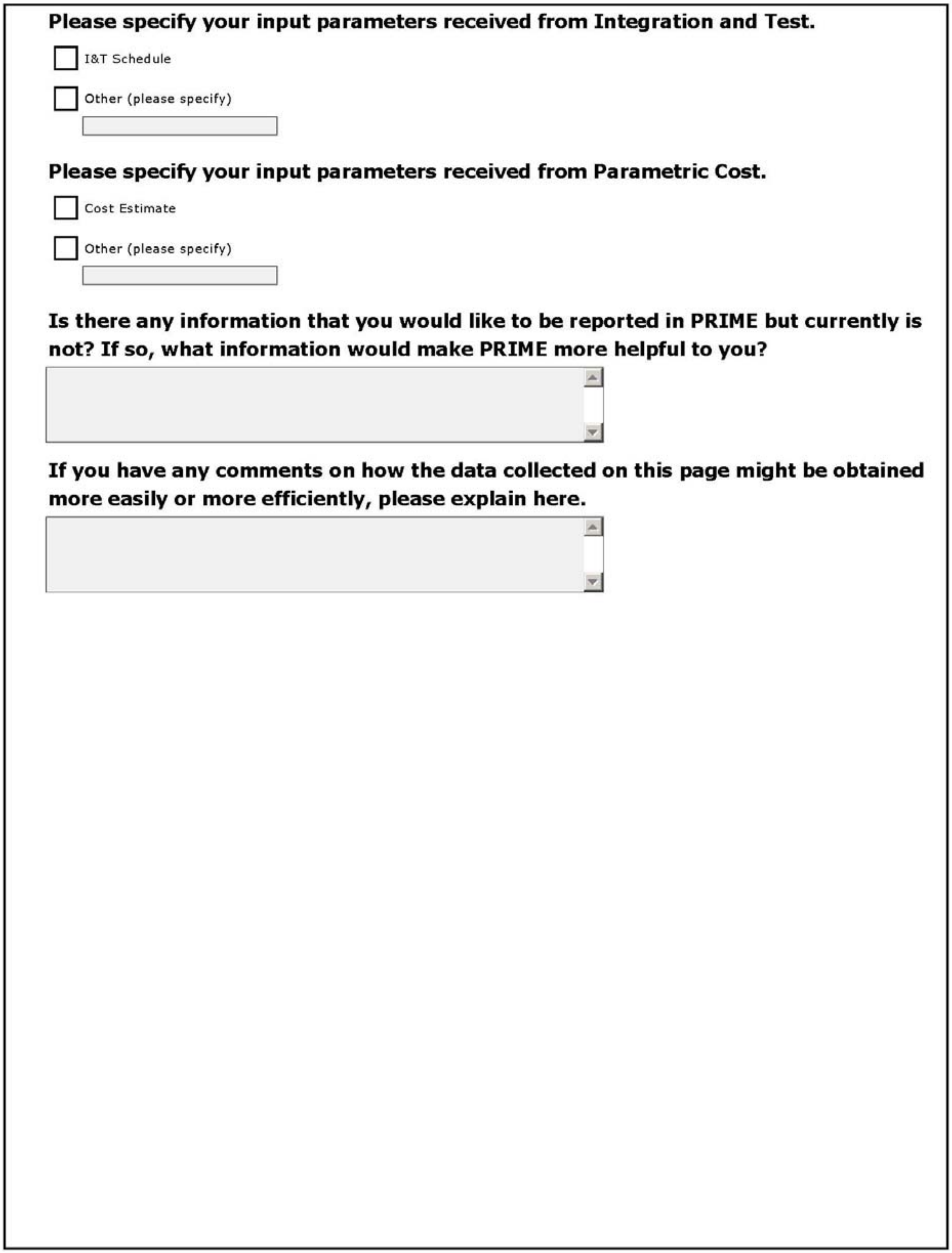
Please specify your input parameters received from Radiation.

☐ Radiation Environment
☐ Other (please specify)

Please specify your input parameters received from Reliability.

☐ Mission Reliability
☐ Subsystem Reliability
☐ Other (please specify)
☐ Redundancy Tradeoffs

Sample Post-Session Survey (Cont.)



Please specify your input parameters received from Integration and Test.

☐ I&T Schedule

☐ Other (please specify)

Please specify your input parameters received from Parametric Cost.

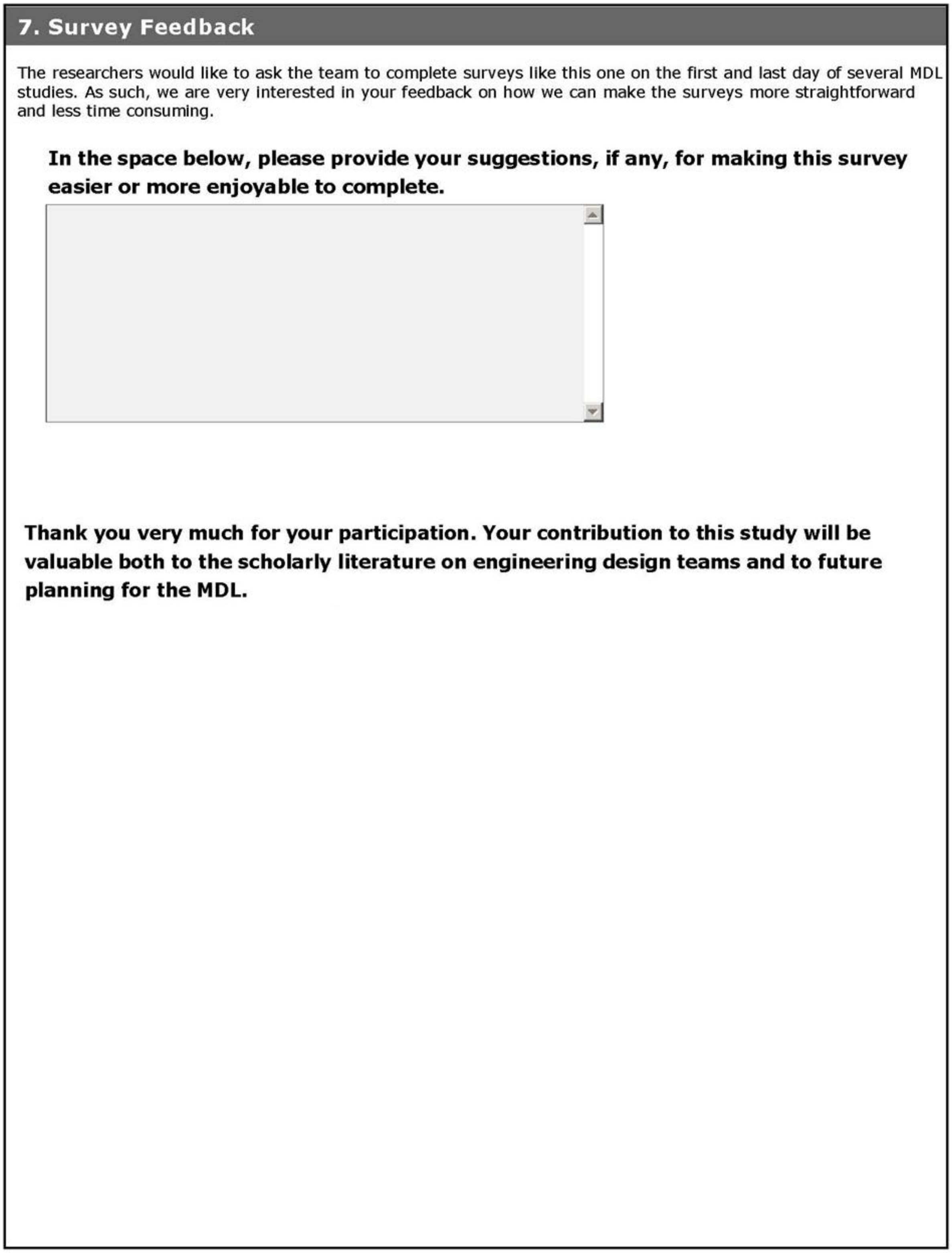
☐ Cost Estimate

☐ Other (please specify)

Is there any information that you would like to be reported in PRIME but currently is not? If so, what information would make PRIME more helpful to you?

If you have any comments on how the data collected on this page might be obtained more easily or more efficiently, please explain here.

Sample Post-Session Survey (Cont.)



7. Survey Feedback

The researchers would like to ask the team to complete surveys like this one on the first and last day of several MDL studies. As such, we are very interested in your feedback on how we can make the surveys more straightforward and less time consuming.

In the space below, please provide your suggestions, if any, for making this survey easier or more enjoyable to complete.

Thank you very much for your participation. Your contribution to this study will be valuable both to the scholarly literature on engineering design teams and to future planning for the MDL.

Appendix B

Mathematical Formalism of the DSM

The purpose of this appendix is to describe the mathematical basis for the parameter-based Design Structure Matrix (DSM). Because of the mathematical construction, the DSM is shown here with 1s instead of Xs to denote marks in the cells.

B.1. Description of the DSM

Consider the following 12 *x* 12 binary matrix:

$$D=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{B-1}$$

Each cell in this matrix can be conceptualized as a representation of a dependency between design parameters. In the convention of the DSM, a 1 is placed in cell *i,j* if and only if parameter X_i requires parameter X_j as an input. So, a 1 in cell *i,j* indicates that

$$X_i = f(X_j) \ . \tag{B-2}$$

In the DSM, the i^{th} row vector is the set of inputs to X_i, and the i^{th} column vector is the set of parameters to which X_i serves as an input. In the matrix, a value of 1 represents the existence of a dependency, and a value of 0 indicates its absence. For example, the set of inputs to parameter X_4 is

$$Input(X_4) = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{B-3}$$

Thus, X_4 takes X_2, X_5, and X_8 as inputs. Formally, Eq. (B-3) implies that

$$X_4 = f(X_2, X_5, X_8). \tag{B-4}$$

Similarly, the set of parameters to which X_4 is provided is

$$Output(X_4) = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}^T. \tag{B-5}$$

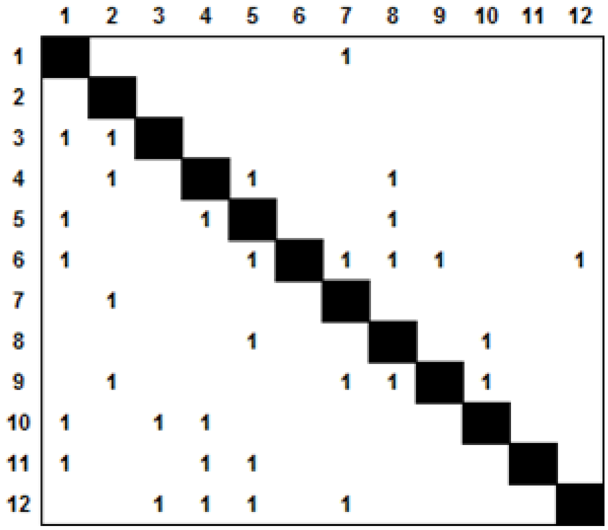
Thus, X_4 is an input to X_5, X_{10}, X_{11}, and X_{12}. So,

$$\begin{aligned} X_4 &= f^{-1}(X_5) \\ X_4 &= f^{-1}(X_{10}) \\ X_4 &= f^{-1}(X_{11}) \\ X_4 &= f^{-1}(X_{12}) \end{aligned}. \tag{B-6}$$

For simplicity, the DSM is presented in this appendix as shown in Figure B-1. The figure is the same matrix, *D*, as described by Eq. (B-1). The 0s have been replaced with white space, and the diagonal has been blocked to indicate that those cells hold no meaning in the DSM because they each represent a parameter's dependence on itself. For the purposes of DSM analysis, the values along the diagonal are taken to be 0.

B.2. Partitioning the DSM

Partitioning is the reordering of rows and columns of a DSM in such a way that reduces feedback and rework to the greatest extent possible. In partitioning, the rows and columns of the DSM are moved together, i.e., if row *i* is moved to row position *j*, column *i* must be moved to



	1	2	3	4	5	6	7	8	9	10	11	12
1	■						1					
2		■										
3	1	1	■									
4		1		■	1			1				
5	1			1	■			1				
6	1				1	■	1	1	1			1
7		1					■					
8					1			■		1		
9		1					1	1	■	1		
10	1		1	1						■		
11	1			1	1						■	
12			1	1	1		1					■

Figure B-1. Example of a Binary Design Structure Matrix.

column position *j*. The rows and columns are arranged so that the resultant matrix is lower triangular – or as close to it as possible. This is done by finding the order that minimizes the number of 1s above the diagonal. This number, F_{tot}, is given by

$$F_{tot} = \sum_{i<j}^{N(N-1)} D[i,j], \tag{B-7}$$

where N is the number of rows/columns in D and $D[i,j]$ is the value of cell i,j.

Once the matrix has been optimally resequenced, or partitioned, the resultant matrix provides the order in which parameters should be computed. The result of partitioning the example matrix D is shown in Figure B-2. In the partitioned DSM, three types of parameters can be discerned: series, parallel, and coupled (Eppinger 1991). *Series* parameters are those that are computed in sequence such that

$$X_{i+1} = f(X_i). \tag{B-8}$$

Parallel tasks are those that can be computed concurrently because neither depends on the other, such that

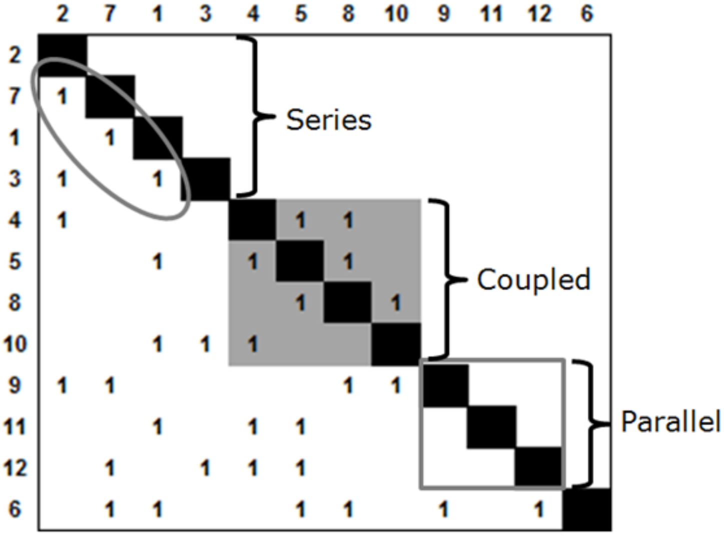


Figure B-2. Partitioning a Binary DSM.

$$\begin{aligned} X_j &\neq f(X_i) \\ X_i &\neq f(X_j) \end{aligned} . \tag{B-9}$$

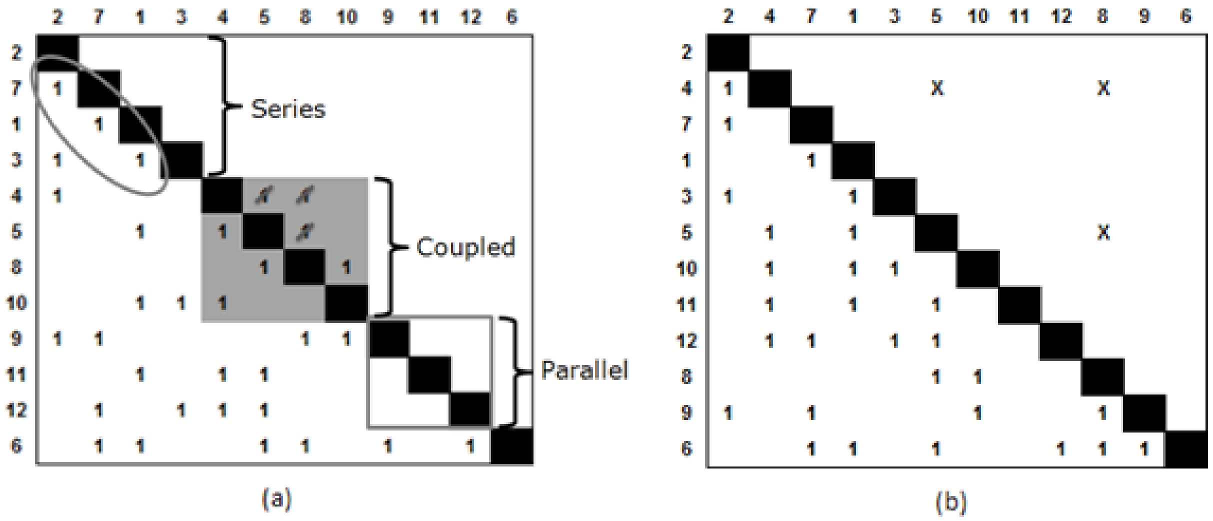
Coupled parameters are those that must be computed together. This situation occurs when it is not possible to resequence the rows and columns in a way that completely eliminates feedback and rework, i.e., when $F_{tot} > 0$ in the partitioned DSM. For example, consider a pair of mutually dependent parameters such that $X_i = f(X_j)$ and $X_j = f(X_i)$. In this case,

$$D[i, j] = D[j, i] = 1 \tag{B-10}$$

regardless of the order of the rows and columns in D, so these two parameters are directly coupled to each other. In addition, parameters can be coupled in more complex ways, such as

$$\begin{aligned} X_i &= f(X_j) \\ X_j &= f(X_k) . \\ X_k &= f(X_i) \end{aligned} \tag{B-11}$$

Of course, this chain of coupling can extend to any number of parameters up to N. Any such collection of interdependent parameters is referred to as a coupled block. In this thesis, all



(a)

	2	7	1	3	4	5	8	10	9	11	12	6
2	■											
7	1	■										
1		1	■									
3	1		1	■								
4	1				■	~~1~~	~~1~~					
5			1		1	■	~~1~~					
8						1	■	1				
10			1	1	1			■				
9	1	1					1	1	■			
11			1		1	1				■		
12		1		1	1	1					■	
6		1	1			1	1		1		1	■

(b)

	2	4	7	1	3	5	10	11	12	8	9	6
2	■											
4	1	■				X				X		
7	1		■									
1			1	■								
3	1			1	■							
5		1		1		■				X		
10		1		1	1		■					
11		1		1		1		■				
12		1	1		1	1			■			
8						1	1			■		
9	1		1				1			1	■	
6			1	1		1			1	1	1	■

Figure B-3. Tearing a Binary DSM. (a) Tearing. (b) Repartitioning. The cells marked with an X indicate starting assumptions in the torn and repartitioned DSM.

partitioning is done using the Excel add-in DSM@MIT (Cho et al. 2004). The general partitioning algorithm is described by Gebala and Eppinger (1991).

B.3. Tearing the DSM

Tearing the DSM reveals the dependencies that, if removed, would yield a sequential process. Mathematically, the goal of tearing is to remove all dependencies above the diagonal so that the matrix is lower-triangular such that

$$F_{tot} = \sum_{i<j}^{N(N-1)} D[i,j] = 0 \, . \tag{B-12}$$

The process of tearing the DSM is shown in Figure B-3. Gebala and Eppinger (1991) describe two methods for identifying the marks to be torn from the DSM. One is based on the judgment of the manager or engineer analyzing the DSM, while the other is done by path searching to quantify and minimize the number of tears made. The procedure used in this thesis is based on patterns of related input parameters in the DSM and subsequent engineering judgment regarding the dependencies that constitute design budgets. After the tearing is completed, the DSM is repartitioned using the DSM@MIT add-in (Cho et al. 2004).

B.4. Clustering the DSM

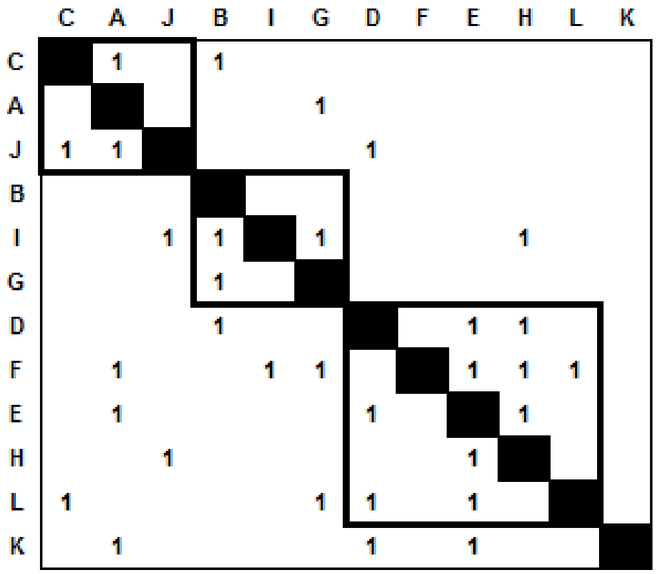
The detailed mathematics behind clustering using the Newman-Girvan community structure algorithm is outside the scope of this discussion. Essentially, it is based on the notion of edge betweenness, which is the number of shortest paths between pairs of nodes in a network on which a given edge falls. The procedure involves computing the betweenness of all edges in the network, removing the edge with the highest betweenness, recalculating betweenness, and repeating until all clusters, or communities have been found (Newman and Girvan 2004). Several possible community structures exist for most networks, but the optimal structure can be found by calculating the value of *modularity* for each community structure.

Modularity is defined as

$$Q = \sum_{z}^{n_c} \left(e_{zz} - a_z^2 \right) \tag{B-13}$$

$$e_{zz} = \frac{m_{zz}}{m},\ a_z = \frac{m_z}{m}\ ,$$

where n_c is the number of communities determined by the algorithm, m is the total number of edges in the network, m_{zz} is the number of edges entirely within the z^{th} community, and m_z is the number of edges to or from any element in the z^{th} community to any other element in the network. Thus, $Q \in [0,1]$ is the proportion of dependencies in the network that are internal to the communities (e_{zz}) adjusted according to the same ratio computed without consideration to community structure (a_z^2). The value of n_c that maximizes Q corresponds to the optimal community structure. For most real-world systems, Q generally is between 0.3 and 0.7 (Newman and Girvan 2004). The result of clustering the example matrix D is shown in Figure B-4. All clustering in this thesis is done using the implementation of the Newman-Girvan algorithm in the UCINET software package (Borgatti et al. 2002).



	C	A	J	B	I	G	D	F	E	H	L	K
C	■	1		1								
A		■				1						
J	1	1	■				1					
B				■								
I			1	1	■	1				1		
G				1		■						
D				1			■		1	1		
F		1			1	1		■	1	1	1	
E		1					1		■	1		
H			1						1	■		
L	1					1	1		1		■	
K		1					1		1			■

Figure B-4. Clustering a Binary DSM.

[This page intentionally left blank]

Appendix C

Loops in the DSM of Space Mission Design

The purpose of this appendix is to provide a full listing of all 187 loops of five parameters or less in the Design Structure Matrix (DSM) representation of the Integrated Concurrent Engineering (ICE) design process. There are 14 three-parameter loops, 51 four-parameter loops, and 122 five-parameter loops. In the format in which the loops are enumerated here, the last parameter in each loop links back to the first parameter in that loop.

In Chapter 4, the shortest loops in the DSM were reduced to 13 types that demonstrate the critical design trades in the process. In reality, it would not be feasible to find all of the millions or even billions of loops in the DSM. For this research, Joseph Kirk provided a modified version of his original Matlab routine, *Count Loops in a Graph* (2007), that was able to find the loops shown here as well as 208 six-parameter loops, 717 seven-parameter loops, and 1,967 eight-parameter loops. This appendix lists just a subset of the 3,079 loops that were found, which is itself a small portion of the many more longer loops that could not be found given the processing power and time available.

The large number of loops in the DSM can be reduced to the 13 types shown in Table 4-3 because of the conceptual similarity among so many sets of loops. Most of the loops listed below repeat essentially the same design trades as described by the 13 loop types but with small changes. In some cases, one or more parameters are different but related. In other cases, some parameters are added to a shorter loop without affecting the basic design trade represented. Still, this appendix is intended to present a more complete (albeit more complex) picture than that provided by the 13 reduced loop types.

C.1. The 14 Three-Parameter Loops

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Mech Mechanisms/Deployables]

[Avionics Processor] → [SW Development and Testing Approaches] → [SW Key Functions]

[Mech Total SS Masses] → [Prop Number and Types of Tanks] → [Prop Mass]

[Mech Structure Material] → [Prop Number and Types of Tanks] → [Orb Debris Casualty Area]

[Mech Mass] → [Orb Debris Casualty Area] → [Mech Structure Material]

[Mech Total SS Masses] → [Prop Number and Types of Thrusters] → [Prop Mass]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Contacts]

[Avionics Configuration / Functionality] → [EPS PSE] → [Avionics Interfaces]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Data Rates]

[EPS Day Power] → [EPS Total System Day Power] → [EPS S/A Drives]

[EPS Night Power] → [EPS Total System Night Power] → [EPS S/A Drives]

[EPS Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives]

[EPS Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives]

[EPS Launch Power] → [EPS Total System Launch Power] → [EPS S/A Drives]

C.2. The 51 Four-Parameter Loops

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm Contacts]

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm Data Rates]

[Mech Total SS Masses] → [Prop Types] → [Prop Number and Types of Tanks] → [Prop Mass]

[Mech Total SS Masses] → [Prop Types] → [Prop Pressure Transducers] → [Prop Mass]

[EPS Total System Day Power] → [Thermal Temperatures, Other Requirements] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Day Power]

[EPS Total System Night Power] → [Thermal Temperatures, Other Requirements] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Night Power]

[EPS Total System Safehold Power] → [Thermal Temperatures, Other Requirements] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Safehold Power]

[EPS Total System Peak Power] → [Thermal Temperatures, Other Requirements] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Peak Power]

[EPS Total System Launch Power] → [Thermal Temperatures, Other Requirements] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Launch Power]

[EPS Total System Day Power] → [Thermal Temperatures, Other Requirements] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Night Power] → [Thermal Temperatures, Other Requirements] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Safehold Power] → [Thermal Temperatures, Other Requirements] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power]

[EPS Total System Peak Power] → [Thermal Temperatures, Other Requirements] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Launch Power] → [Thermal Temperatures, Other Requirements] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[Mech S/C Dimensions] → [Prop Number and Types of Thrusters] → [Prop Number and Types of Components] → [Mech Component Placement]

[Mech S/C Dimensions] → [Prop Number and Types of Tanks] → [Prop Number and Types of Components] → [Mech Component Placement]

[Mech S/C Dimensions] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Number and Types of Components] → [Mech Component Placement]

[Mech Total SS Masses] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Prop Mass]

[Mech Total SS Masses] → [Prop Number and Types of Tanks] → [Prop Valves (Latch, Flow Control)] → [Prop Mass]

[Mech Total SS Masses] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass]

[ACS Mass] → [Mech Total SS Masses] → [ACS Type] → [ACS Sensors]

[ACS Mass] → [Mech Total SS Masses] → [ACS Type] → [ACS Actuators]

[ACS Type] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses]

[ACS Mass] → [Mech Total SS Masses] → [ACS Modes] → [ACS Sensors]

[ACS Mass] → [Mech Total SS Masses] → [ACS Modes] → [ACS Actuators]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses]

[EPS Night Power] → [EPS Total System Night Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE]

[EPS Peak Power] → [EPS Total System Peak Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE]

[EPS Launch Power] → [EPS Total System Launch Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE]

[EPS Total System Night Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Peak Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Launch Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[Avionics Configuration / Functionality] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces]

[Mech Total SS Masses] → [Prop Number and Types of Thrusters] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass]

[EPS Total System Day Power] → [EPS S/A Drives] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Night Power] → [EPS S/A Drives] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Safehold Power] → [EPS S/A Drives] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power]

[EPS Total System Peak Power] → [EPS S/A Drives] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Launch Power] → [EPS S/A Drives] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[Mech Total SS Masses] → [Prop Number and Types of Thrusters] → [Prop Valves (Latch, Flow Control)] → [Prop Mass]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Ground Stations/Antennas] → [Comm Contacts]

[EPS Day Power] → [EPS Total System Day Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE]

[EPS Night Power] → [EPS Total System Night Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE]

[EPS Peak Power] → [EPS Total System Peak Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE]

[Avionics Day Power] → [EPS PSE] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Night Power] → [EPS PSE] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Peak Power] → [EPS PSE] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Safehold Power] → [EPS PSE] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Launch Power] → [EPS PSE] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Ground Stations/Antennas] → [Comm Data Rates]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm S/C Antennas] → [Comm Data Rates]

C.3. The 122 Five-Parameter Loops

[Flt Dyn Orbital Parameters] → [Comm Orbit Determination] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Flt Dyn Viewing Periods (Eclipse Times)]

[Flt Dyn Viewing Periods (Eclipse Times)] → [Comm Contacts] → [Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas]

[Flt Dyn Viewing Periods (Eclipse Times)] → [Comm Contacts] → [Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Ground Stations/Antennas]

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm S/C Antennas] → [Comm Data Rates]

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm S/C Antennas] → [Mech Mechanisms/Deployables]

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm Contacts] → [Avionics Mass Data Storage]

[Ops Concept] → [Comm Ground Station Cost Per Year] → [Comm Ground Stations/Antennas] → [Comm Data Rates] → [Avionics Mass Data Storage]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Mech Mechanisms/Deployables] → [ACS Type] → [ACS Sensors]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Mech Mechanisms/Deployables] → [ACS Modes] → [ACS Sensors]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Day Power] → [EPS Total System Day Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Night Power] → [EPS Total System Night Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Safehold Power] → [EPS Total System Safehold Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Peak Power] → [EPS Total System Peak Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Launch Power] → [EPS Total System Launch Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power] → [EPS Total System Day Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power] → [EPS Total System Night Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power] → [EPS Total System Safehold Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power] → [EPS Total System Peak Power] → [Thermal Temperatures, Other Requirements]

[Ops Concept] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power] → [EPS Total System Launch Power] → [Thermal Temperatures, Other Requirements]

[ACS Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia] → [ACS Type] → [ACS Sensors]

[ACS Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia] → [ACS Type] → [ACS Actuators]

[ACS Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia] → [ACS Modes] → [ACS Sensors]

[ACS Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia] → [ACS Modes] → [ACS Actuators]

[ACS Day Power] → [EPS Total System Day Power] → [EPS S/A Drives] → [ACS Type] → [ACS Sensors]

[ACS Day Power] → [EPS Total System Day Power] → [EPS S/A Drives] → [ACS Type] → [ACS Actuators]

[ACS Day Power] → [EPS Total System Day Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Sensors]

[ACS Day Power] → [EPS Total System Day Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Actuators]

[ACS Night Power] → [EPS Total System Night Power] → [EPS S/A Drives] → [ACS Type] → [ACS Sensors]

[ACS Night Power] → [EPS Total System Night Power] → [EPS S/A Drives] → [ACS Type] → [ACS Actuators]

[ACS Night Power] → [EPS Total System Night Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Sensors]

[ACS Night Power] → [EPS Total System Night Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Actuators]

[ACS Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives] → [ACS Type] → [ACS Sensors]

[ACS Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives] → [ACS Type] → [ACS Actuators

[ACS Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Sensors]

[ACS Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Actuators]

[ACS Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives] → [ACS Type] → [ACS Sensors]

[ACS Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives] → [ACS Type] → [ACS Actuators]

[ACS Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Sensors]

[ACS Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Actuators]

[ACS Launch Power] → [EPS Total System Launch Power] → [EPS S/A Drives] → [ACS Type] → [ACS Sensors]

[ACS Launch Power] → [EPS Total System Launch Power] → [EPS S/A Drives] → [ACS Type] → [ACS Actuators]

[ACS Launch Power] → [EPS Total System Launch Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Sensors]

[ACS Launch Power] → [EPS Total System Launch Power] → [EPS S/A Drives] → [ACS Modes] → [ACS Actuators]

[ACS Type] → [ACS Sensors] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass] → [Mech Total SS Masses]

[ACS Type] → [ACS Actuators] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses]

[ACS Type] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia]

[ACS Type] → [Prop Number and Types of Thrusters] → [Prop Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives]

[ACS Type] → [Prop Number and Types of Thrusters] → [Prop Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives]

[ACS Type] → [Prop Number and Types of Thrusters] → [Prop Valves (Latch, Flow Control)] → [Prop Mass] → [Mech Total SS Masses]

[ACS Type] → [Prop Number and Types of Thrusters] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass] → [Mech Total SS Masses]

[ACS Sensors] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass] → [Mech Total SS Masses] → [ACS Modes]

[ACS Actuators] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses] → [ACS Modes]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Prop Mass] → [Mech Total SS Masses] → [Mech Moments of Inertia]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Prop Safehold Power] → [EPS Total System Safehold Power] → [EPS S/A Drives]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Prop Peak Power] → [EPS Total System Peak Power] → [EPS S/A Drives]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Prop Valves (Latch, Flow Control)] → [Prop Mass] → [Mech Total SS Masses]

[ACS Modes] → [Prop Number and Types of Thrusters] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass] → [Mech Total SS Masses]

[Avionics Mass] → [Mech Total SS Masses] → [Prop Number and Types of Thrusters] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Day Power] → [EPS Total System Day Power] → [EPS Number of Strings] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Day Power] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Night Power] → [EPS Total System Night Power] → [EPS Number of Strings] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Night Power] → [EPS Total System Night Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Night Power] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Peak Power] → [EPS Total System Peak Power] → [EPS Number of Strings] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Peak Power] → [EPS Total System Peak Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Peak Power] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Safehold Power] → [EPS Total System Safehold Power] → [EPS Number of Strings] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Safehold Power] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Launch Power] → [EPS Total System Launch Power] → [EPS Number of Strings] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Launch Power] → [EPS Total System Launch Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Launch Power] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Avionics Interfaces] → [Avionics Configuration / Functionality]

[Avionics Processor] → [Avionics Duty Cycle and Percentage of Mission Life in Use] → [Rel Need for Redundancy] → [SW Development and Testing Approaches] → [SW Key Functions]

[Avionics Processor] → [Avionics Redundancy of Components] → [Rel Need for Redundancy] → [SW Development and Testing Approaches] → [SW Key Functions]

[Avionics Mass Data Storage] → [Comm Data Volumes, Compression] → [Comm Ground Stations/Antennas] → [Comm S/C Antennas] → [Comm Data Rates]

[Comm S/C Antennas] → [Comm Number and Types of Components] → [Mech Component Placement] → [Mech S/C Dimensions] → [Mech Structure Shape]

[EPS Day Power] → [EPS Total System Day Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Day Power] → [EPS Total System Day Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Night Power] → [EPS Total System Night Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Night Power] → [EPS Total System Night Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Peak Power] → [EPS Total System Peak Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Peak Power] → [EPS Total System Peak Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Safehold Power] → [EPS Total System Safehold Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Launch Power] → [EPS Total System Launch Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [EPS Array Mass]

[EPS Total System Day Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Day Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Day Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Day Power] → [EPS S/A Drives] → [Mech Mechanisms/Deployables] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Day Power] → [Thermal Temperatures, Other Requirements] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Day Power]

[EPS Total System Night Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Night Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Night Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Night Power] → [EPS S/A Drives] → [Mech Mechanisms/Deployables] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Night Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Night Power] → [Thermal Temperatures, Other Requirements] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Night Power]

[EPS Total System Peak Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Peak Power] → [EPS Maximum Average Load, Bus Voltage] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Peak Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Peak Power] → [EPS S/A Drives] → [Mech Mechanisms/Deployables] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Peak Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Peak Power] → [Thermal Temperatures, Other Requirements] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Peak Power]

[EPS Total System Safehold Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power]

[EPS Total System Safehold Power] → [EPS S/A Drives] → [Mech Mechanisms/Deployables] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power]

[EPS Total System Safehold Power] → [Thermal Temperatures, Other Requirements] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Safehold Power]

[EPS Total System Launch Power] → [EPS Number of Strings] → [EPS Array (Area, Cells, Efficiencies)] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[EPS Total System Launch Power] → [EPS S/A Drives] → [Mech Mechanisms/Deployables] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[EPS Total System Launch Power] → [EPS Battery (Type, Mass, Depth of Discharge)] → [EPS PSE] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[EPS Total System Launch Power] → [Thermal Temperatures, Other Requirements] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Launch Power]

[Mech Mass] → [Mech Total SS Masses] → [Prop Number and Types of Tanks] → [Orb Debris Casualty Area] → [Mech Structure Material]

[Mech Total SS Masses] → [Prop Types] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Prop Mass]

[Mech Total SS Masses] → [Prop Types] → [Prop Number and Types of Tanks] → [Prop Valves (Latch, Flow Control)] → [Prop Mass]

[Mech Total SS Masses] → [Prop Types] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass]

[Mech Total SS Masses] → [Prop Number and Types of Thrusters] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Prop Mass]

[Mech Total SS Masses] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Prop Valves (Latch, Flow Control)] → [Prop Mass]

[Mech Total SS Masses] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Mass]

[Mech Structure Shape] → [Prop Number and Types of Thrusters] → [Prop Number and Types of Components] → [Mech Component Placement] → [Mech S/C Dimensions]

[Mech Structure Shape] → [Prop Number and Types of Tanks] → [Prop Number and Types of Components] → [Mech Component Placement] → [Mech S/C Dimensions]

[Mech Structure Shape] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Number and Types of Components] → [Mech Component Placement] → [Mech S/C Dimensions]

[Mech Structure Material] → [Prop Number and Types of Thrusters] → [Prop Propellant Quantities + Gas Mass] → [Prop Number and Types of Tanks] → [Orb Debris Casualty Area]

[Mech Mechanisms/Deployables] → [Mech Number and Types of Components] → [Mech Component Placement] → [Mech S/C Dimensions] → [Thermal Radiators (sizes, placements, types, margins)]

[Mech S/C Dimensions] → [Mech Cross-Sectional Area] → [Thermal Radiators (sizes, placements, types, margins)] → [Thermal Number and Types of Components] → [Mech Component Placement]

[Mech S/C Dimensions] → [Prop Number and Types of Thrusters] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Number and Types of Components] → [Mech Component Placement]

[Mech S/C Dimensions] → [Prop Number and Types of Tanks] → [Thermal Heaters, Controllers, Thermistors] → [Thermal Number and Types of Components] → [Mech Component Placement]

Appendix D

Shared Mental Model Networks

This appendix shows the structure of the shared mental model networks for each of the 12 design sessions observed in the Mission Design Laboratory (MDL). Each individual network describes a team mental model at a snapshot in time. In the thesis, a measure of structural similarity between the pre- and post-session network for each design session was used to quantify dynamics of shared knowledge in the team. A broader network analysis can be conducted on each of the 24 networks to understand specific elements of the static structure of the team mental model that each network represents. This detailed analysis of mental models in individual design sessions is one possible area for future work. To facilitate this, the network graphs shown here are color coded. Red nodes represent the discipline engineers on the design team, blue nodes represent the Team Lead and Systems Engineer(s), and green nodes represent the customer team.

In the caption for each figure, three quantities measuring shared knowledge in the team are given. The first is the metric for change in shared knowledge, ΔS, computed using the structural approach. The next two quantities are the team mental model for the pre-session network, S_{pre}, and for the post-session network, S_{post}. These quantities are computed as the average of all pair-wise values of sharedness, $S_{x,y}$, and thus are based on the collective approach to shared knowledge. The purpose of showing all three quantities here is to provide the data that leads to the assessment that the metric of change in shared knowledge constitutes an increase in shared knowledge in the team.

The network graphs and the listed metrics provide a description of shared knowledge in the team. If more detail is required, the raw survey data on major design drivers are available from the author upon request by e-mail to avnet@alum.mit.edu.

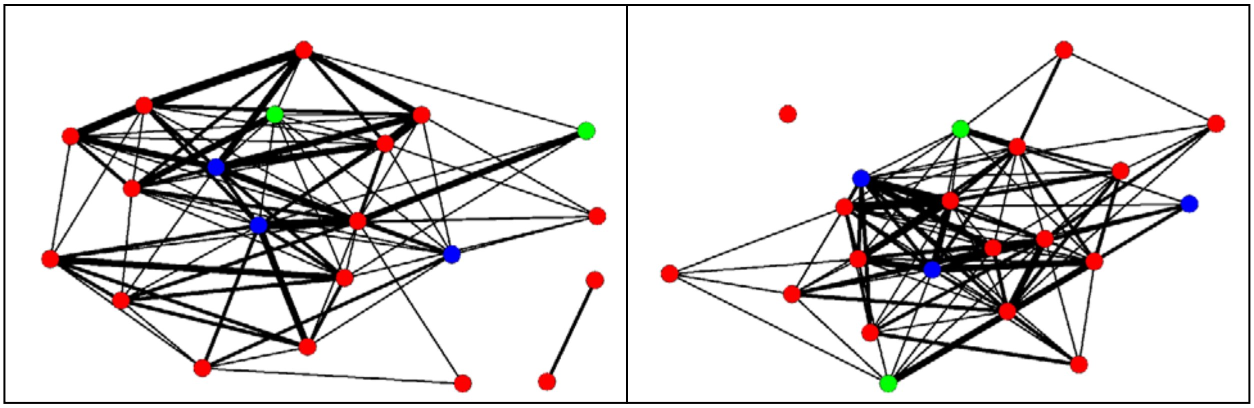

(a) (b)

Figure D-1. Structure of Shared Knowledge for Session 1, ΔS = 0.298. (a) Pre-Session, S_{pre} = 0.387. (b) Post-Session, S_{post} = 0.420.

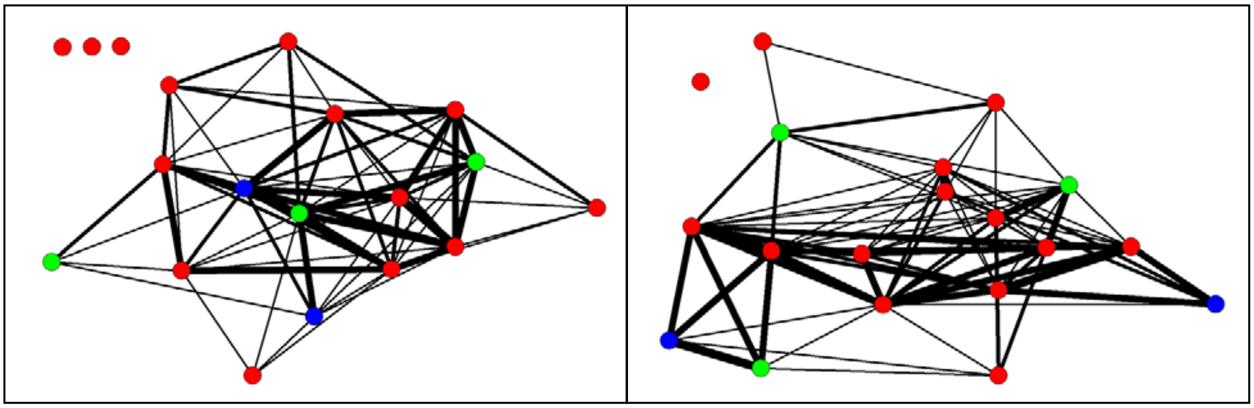
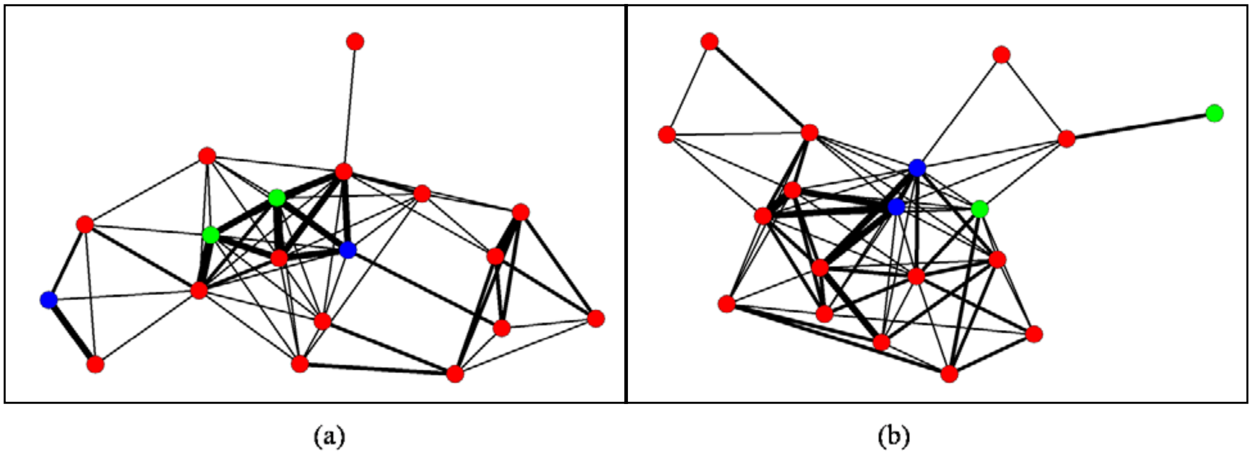
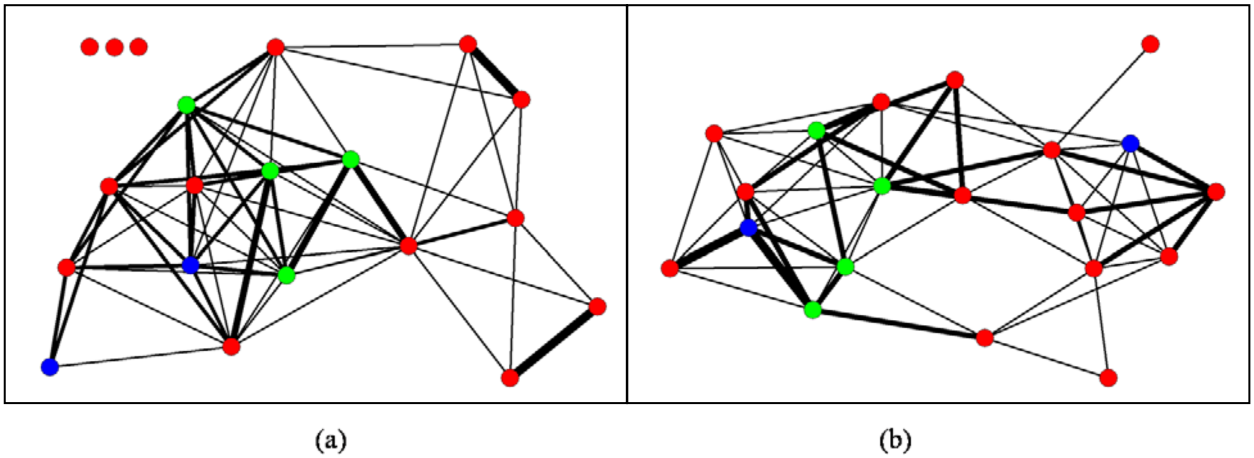

(a) (b)

Figure D-2. Structure of Shared Knowledge for Session 2, ΔS = 0.357. (a) Pre-Session, S_{pre} = 0.341. (b) Post-Session, S_{post} = 0.425.



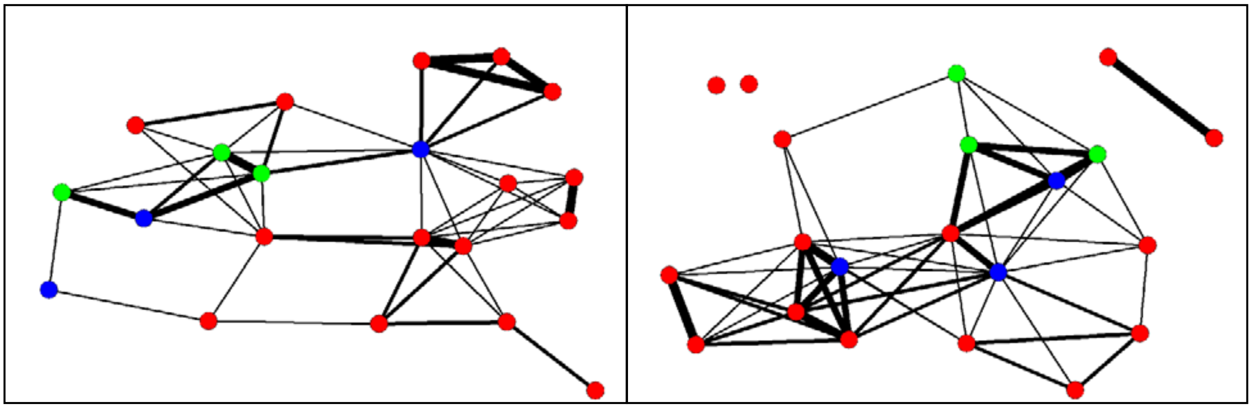
(a) (b)

Figure D-3. Structure of Shared Knowledge for Session 3, ΔS = 0.406. (a) Pre-Session, S_{pre} = 0.348. (b) Post-Session, S_{post} = 0.364.



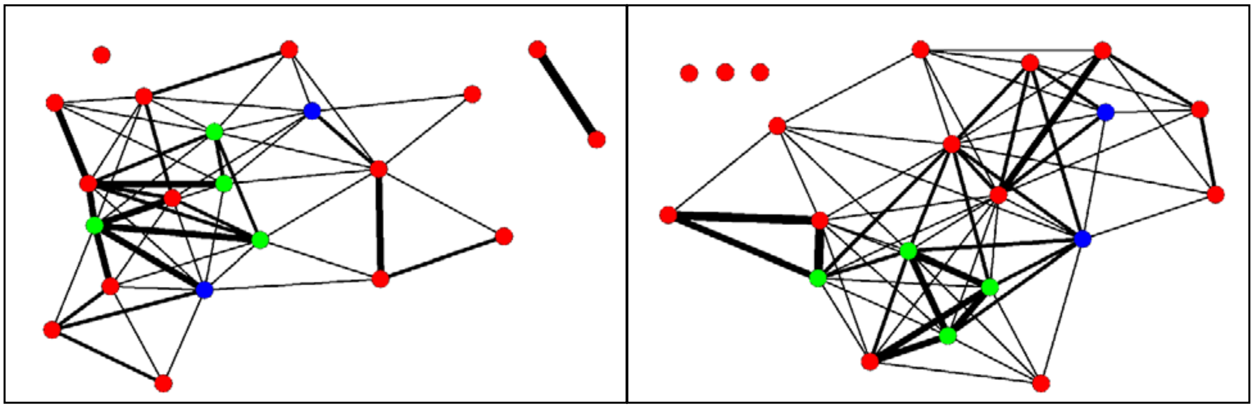
(a) (b)

Figure D-4. Structure of Shared Knowledge for Session 4, ΔS = 0.388. (a) Pre-Session, S_{pre} = 0.289. (b) Post-Session, S_{post} = 0.319.



(a) (b)

Figure D-5. Structure of Shared Knowledge for Session 5, ΔS = 0.356. (a) Pre-Session, S_{pre} = 0.249. (b) Post-Session, S_{post} = 0.267.



(a) (b)

Figure D-6. Structure of Shared Knowledge for Session 6, ΔS = 0.345. (a) Pre-Session, S_{pre} = 0.236. (b) Post-Session, S_{post} = 0.309.

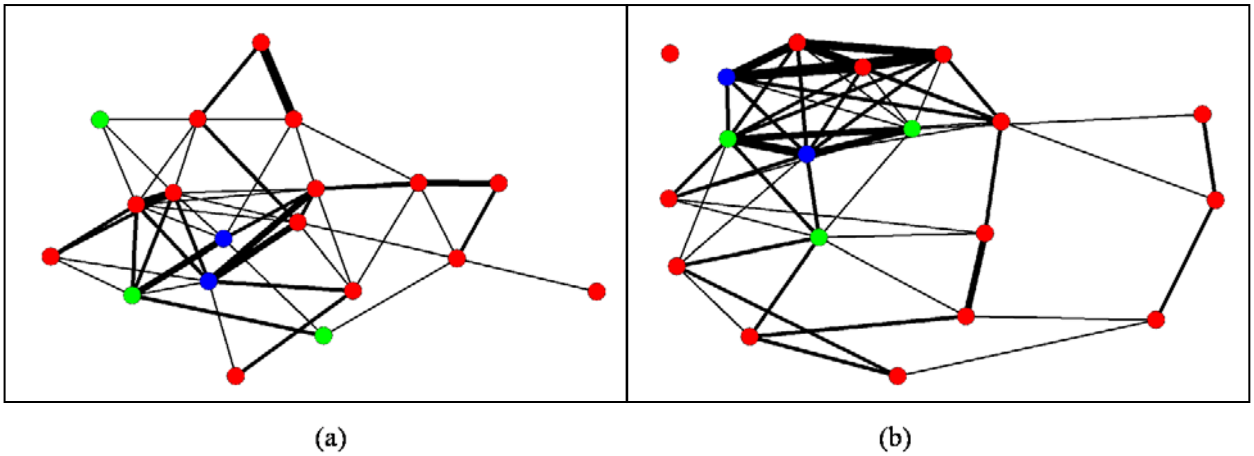


Figure D-7. Structure of Shared Knowledge for Session 7, ΔS = 0.370. (a) Pre-Session, S_{pre} = 0.305. (b) Post-Session, S_{post} = 0.318.

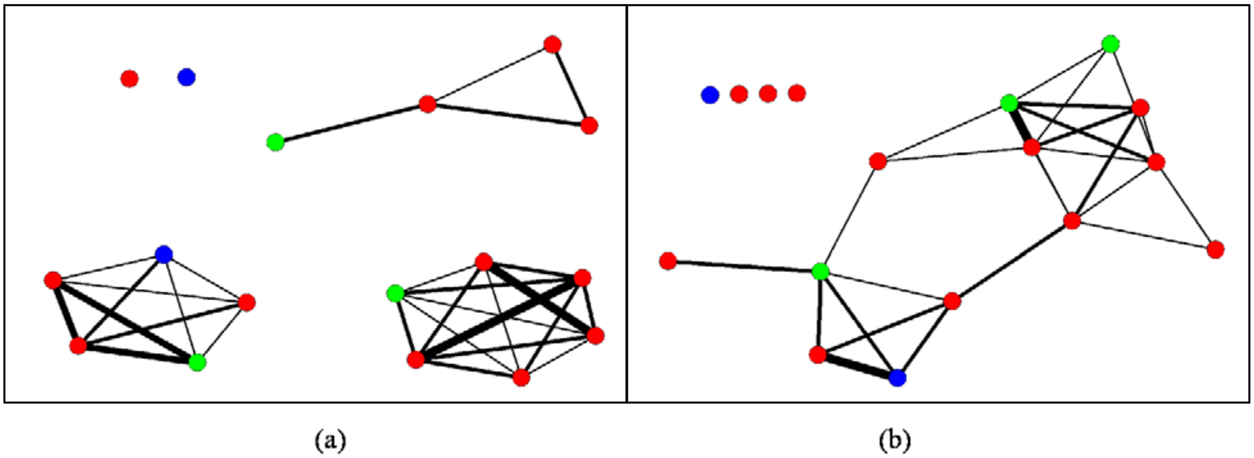


Figure D-8. Structure of Shared Knowledge for Session 8, ΔS = 0.376. (a) Pre-Session, S_{pre} = 0.210. (b) Post-Session, S_{post} = 0.210. This is an exceptional case because $S_{pre} \approx S_{post}$, but it is still in agreement with the finding that $S_{pre} \leq S_{post}$. The structure of these networks is particularly interesting because the post-session network has a single large connected component, but it also includes more isolates than does the pre-session network.

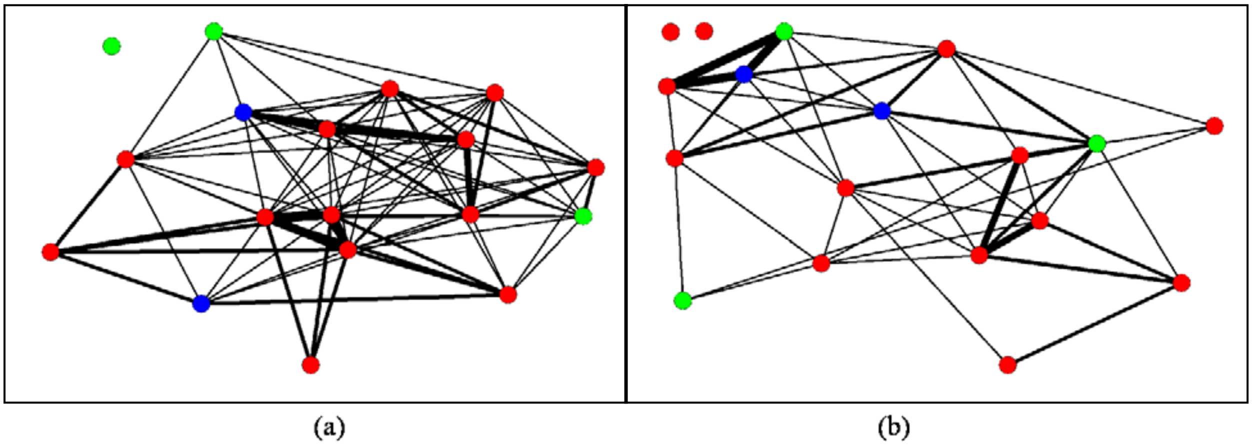


Figure D-9. Structure of Shared Knowledge for Session 9, Δ*S* = 0.506. (a) Pre-Session, S_{pre} = 0413. (b) Post-Session, S_{post} = 0.304. This is the lunar surface operations design session, the only one for which $S_{pre} > S_{post}$. Because Δ*S* is large (corresponding to a QAP correlation $C_{SMM} \approx 0$), the pre-session and post-session structure of shared knowledge show virtually no discernible similarity. Thus, the relationship between S_{pre} and S_{post} is not meaningful in this case.

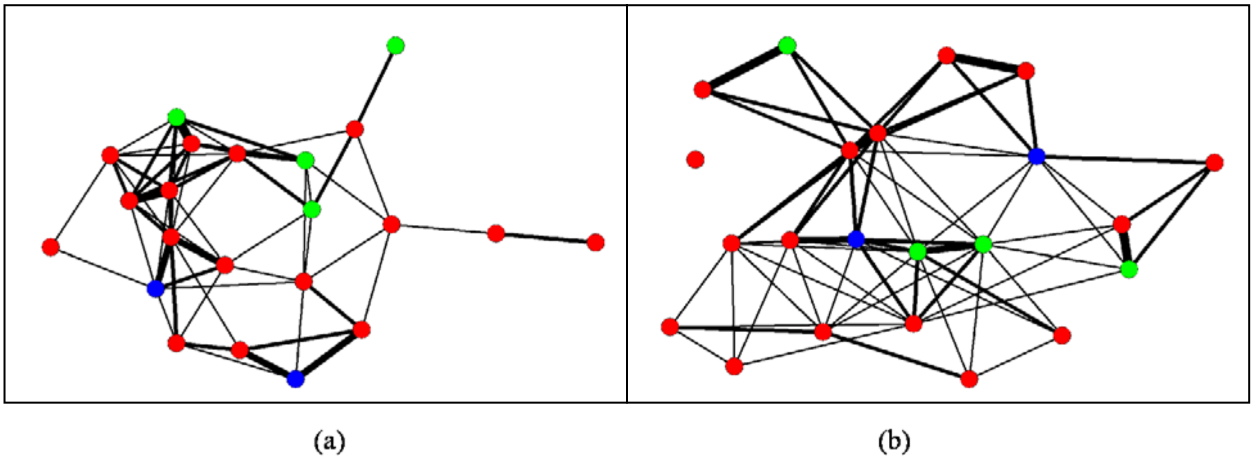


Figure D-10. Structure of Shared Knowledge for Session 10, Δ*S* = 0.453. (a) Pre-Session, S_{pre} = 0.289. (b) Post-Session, S_{post} = 0.299.

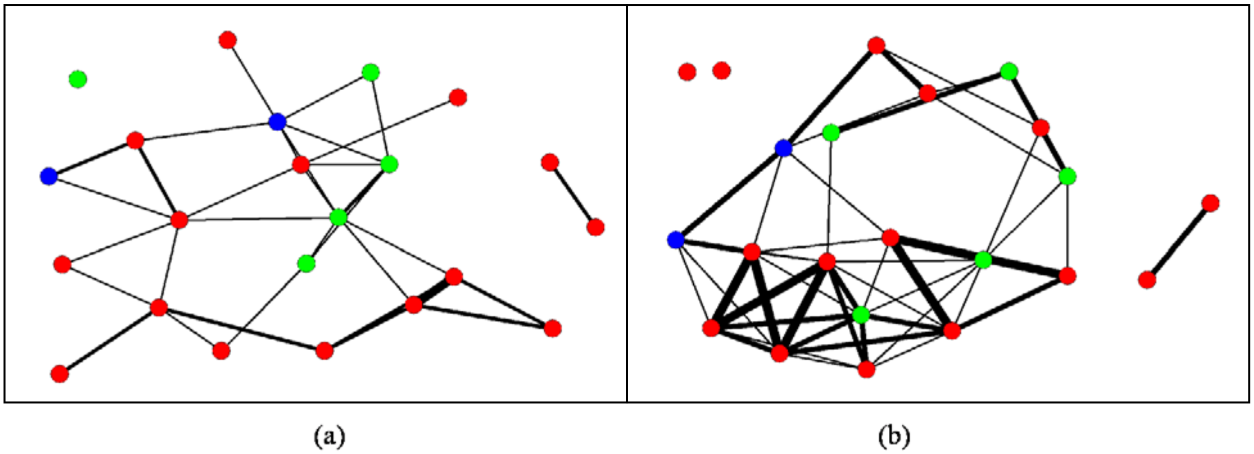

(a) (b)

Figure D-11. Structure of Shared Knowledge for Session 11, Δ*S* = 0.443. (a) Pre-Session, S_{pre} = 0.176. (b) Post-Session, S_{post} = 0.229.

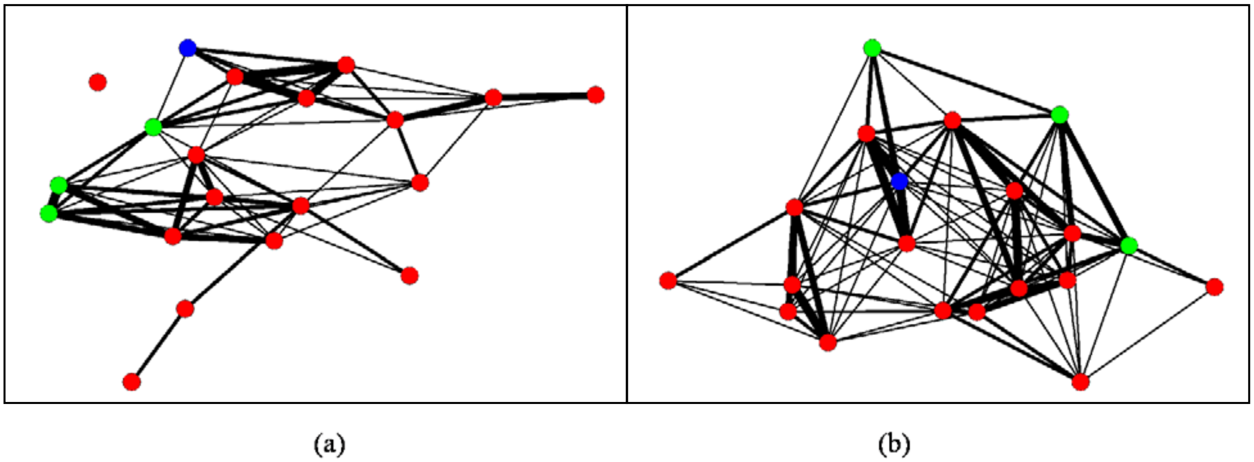

(a) (b)

Figure D-12. Structure of Shared Knowledge for Session 12, Δ*S* = 0.297. (a) Pre-Session, S_{pre} = 0.351. (b) Post-Session, S_{post} = 0.437.

[This page intentionally left blank]

Appendix E

Design Drivers: The Content of Mental Models

The purpose of this appendix is to provide a more complete description of the data on the content of mental models than was necessary or feasible in Chapter 6. The appendix is divided into two sections. In the first section, the correlation matrices representing the content of mental models for the 20 possible design drivers are provided. In the second section, a few interesting non-statistical relationships among the data are presented.

E.1. Design Driver Correlation Matrices

In the discussion on the content of mental models in Chapter 6, it was stated that Communications is the only design driver for which statistically significant correlations exist among all of the relevant metrics: Technology Readiness Level (TRL) of driver j, $I_{P,j\ (Pre)}$ (pre-session perceived importance of driver j), $I_{P,j\ (Post)}$ (post-session perceived importance of driver j), and ΔS (change in shared knowledge in the team). In that chapter, it was stated that among the possible correlations among those metrics, no more than one-third are statistically significant for 19 of the 20 design drivers. For Communications, however, all six correlations are statistically significant. Table E-1 shows the correlation matrix for each of the 20 possible design drivers. In some cells, "N/A" is used to denote that the data were not available – generally because the Technology Readiness Level is not a meaningful metric for many of the drivers (e.g., Reliability, Orbital Debris, and Management). For Flight Software and Thermal, those relationships are not applicable because their TRLs were the same in all 12 design sessions.

Five of the six correlations among the metrics for Communications were presented in the discussion in Chapter 6. The one that was not shown was the correlation between pre-session and post-session perceived importance of the driver. Because that correlation only provides information about the perceived importance of a single driver, it is less important for determining relationships between shared knowledge and the technical system than are the other correlations. Still, it demonstrates an interesting result. The importance of Communications in the team does

Table E-1. Correlation Matrices of Perceived Importance for All Design Drivers. Spearman's ρ is used for all correlations involving TRL.

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	-0.025	-		
$I_{P,j\,(Post)}$	-0.152	0.691*	-	
ΔS	0.077	-0.612	-0.797**	-

Attitude Control (j = Acs)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	-0.153	-		
$I_{P,j\,(Post)}$	-0.495	0.767**	-	
ΔS	-0.621	0.020	-0.003	-

Avionics (j = Avi)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	0.688*	-		
$I_{P,j\,(Post)}$	0.636*	0.964***	-	
ΔS	-0.820**	0.864**	0.741*	-

Communications (j = Comm)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.972***	-	
ΔS	N/A	0.005	-0.164	-

Contamination (j = Cont)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.624	-	
ΔS	N/A	0.141	-0.258	-

Cost (j = Cost)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	0.142	-		
$I_{P,j\,(Post)}$	0.212	0.902***	-	
ΔS	-0.805**	-0.109	-0.199	-

Electrical Power (j = Eps)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.338	-	
ΔS	N/A	0.579	0.334	-

Flight Dynamics (j = Fdyn)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.839**	-	
ΔS	N/A	0.148	-0.284	-

Flight Software (j = Fsw)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.817**	-	
ΔS	N/A	-0.367	-0.415	-

Instruments (j = Inst)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.877***	-	
ΔS	N/A	-0.044	-0.100	-

Integration and Test (j = Intg)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	0.493	-		
$I_{P,j\,(Post)}$	0.620	0.734*	-	
ΔS	-0.890*	0.023	0.419	-

Launch Vehicles (j = Lnch)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.146	-	
ΔS	N/A	-0.089	0.291	-

Management (j = Mgmt)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	0.011	-		
$I_{P,j\,(Post)}$	0.111	0.566	-	
ΔS	-0.311	-0.385	-0.636*	-

Mechanical (j = Mech)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.214	-	
ΔS	N/A	0.716*	0.327	-

Mission Operations (j = Mops)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	-0.167	-	
ΔS	N/A	-0.122	-0.080	-

Orbital Debris (j = Debr)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	0.100	-		
$I_{P,j\,(Post)}$	0.179	0.907***	-	
ΔS	-0.328	0.540	0.590	-

Propulsion (j = Prop)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	-0.151	-	
ΔS	N/A	0.831**	-0.080	-

Radiation (j = Rad)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.676*	-	
ΔS	N/A	0.155	-0.035	-

Reliability (j = Rel)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.818*	-	
ΔS	N/A	-0.247	-0.231	-

Schedule (j = Schd)

	TRL	$I_{P,j\,(Pre)}$	$I_{P,j\,(Post)}$	ΔS
TRL	-			
$I_{P,j\,(Pre)}$	N/A	-		
$I_{P,j\,(Post)}$	N/A	0.001	-	
ΔS	N/A	0.407	-0.102	-

Thermal (j = Thrm)

*** $p < 0.001$ ** $p < 0.01$ * $p < 0.05$

not change significantly over the course of the design session. This may imply that the team already understood the subsystem as well prior to the start of the work as they would at the end, which supports the interpretation that the orbit determination trades involving Communications should be resolved early in each design session. On the other hand, the pre-post importance correlation was statistically significant for 12 of the 20 possible design drivers, so that relationship cannot be used to state any specific findings regarding a particular discipline.

E.2. Other Findings about Design Driver Perceived Importance

The data on the content of mental models show that the perceived importance of the Propulsion subsystem is directly related to the type of mission being designed. This result is not a finding about the relationship between shared knowledge and the technical system, but rather it demonstrates that the team answered the survey question as intended since they apparently related the importance of Propulsion to the amount of propellant needed and the difficulty of maneuvers made. Table E-2 lists the 12 observed design sessions sorted by the pre-session perceived importance of Propulsion, $I_{P,Prop\ (Pre)}$. As the table demonstrates, the three sessions with the highest values of $I_{P,Prop\ (Pre)}$ are the design sessions for missions to other planets in the solar system. The fourth session listed is similar to the first three from a mission dynamics standpoint in that it is in a heliocentric (Sun-centered) orbit and thus is not influenced by Earth's gravity. The fifth and sixth sessions on the list are at the Sun-Earth libration point 2 (L2) and on the lunar surface, respectively. Finally, the last six design sessions listed are all in Earth orbit (two Earth science missions followed by four space science missions).

The ordering of the post-session results is identical to the ordering in the pre-session case except for one small change. The asterisk (*) next to the value of $I_{P,Prop\ (Post)}$ indicates that it is the only session whose position in the list changes from pre-session to post-session. The post-session order places session 11 above 12 in the list (i.e., $I_{P,Prop\ (Post)}$ for session 11 is larger than for session 12). Although this splits the group of planetary missions, the full set of four missions whose destinations are outside the influence of Earth's gravity remain together in the sorted list, and all other groupings are the same as in the pre-session case.

In addition, it should be noted that a related but simpler pattern exists for Flight Dynamics. In the pre-session data, Flight Dynamics was perceived as less important for Earth-orbiting

Table E-2. Propulsion in the Content of Shared Mental Models. Sorting the design sessions by pre-session perceived importance of Propulsion reveals a grouping among the design sessions according to type. This grouping remains when post-session data are used except for the position of session 11, whose post-session perceived importance is indicated by an asterisk (*). Still, the groups remain unchanged if the mission type labeled Planetary is broadened to include any mission outside the influence of Earth's gravity.

	Session Type	$I_{P,Prop}$ Pre-Session	$I_{P,Prop}$ Post-Session
1	Planetary	0.857	0.857
10	Planetary	0.500	0.591
12	Planetary	0.350	0.400
11	Multiple Spacecraft in a Heliocentric Orbit	0.182	0.500*
8	Costing for a Major Program at Sun-Earth L2	0.176	0.235
9	Concept Generation for Lunar Surface Operations	0.111	0.167
3	Earth Science	0.105	0.158
5	Earth Science	0.095	0.143
4	Space Science	0.050	0.050
6	Space Science	0.000	0.048
2	Space Science	0.000	0.000
7	Space Science	0.000	0.000

missions than for other types. In the post-session data, however, that result changed somewhat. No discernible pattern of this type was found for the perceived importance of other possible design drivers.