The Role of Product Development Metrics for Making Design Decisions in the Defense Aerospace Industry

by

Todd Michael Stout
B.S., Aerospace Engineering
University of Kansas, 1992

Submitted to the Department of Mechanical Engineering and the Technology and Policy Program in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Mechanical Engineering

and

Master of Science in Technology and Policy

at the
Massachusetts Institute of Technology
February 1996

Signature of Author ....
Department of Mechanical Engineering
October 18, 1995

Certified by ....
David P. Hoult
Thesis Supervisor

Accepted by ....
Richard de Neufville
Chairman, Technology and Policy Program

Accepted by ....
Ain A. Sonin
Departmental Committee on Graduate Studies

The Role of Product Development Metrics for Making Design Decisions in the Defense Aerospace Industry

by

Todd Michael Stout

Submitted to the Department of Mechanical Engineering and the Technology and Policy Program on October 18, 1995 in partial fulfillment of the requirements for the degrees of Master of Science in Mechanical Engineering and Master of Science in Technology and Policy

Abstract

In current product development activities, many companies are unable to accurately predict the success of their efforts. This leads companies into dead-end development paths and often results in output that meets the contracted requirements for the program but fails to satisfy either the internal or external customers' needs. These problems arise primarily from one or more of three common problems during the development: failure to focus on the proper metrics and measurements of current activities; failure to maintain a significant historical database to facilitate corporate learning; and the use of a decision-making process that often lacks the information necessary to make good decisions.

This thesis identifies these problems through three case studies of product modifications and upgrade development programs in the defense aircraft industry. From these cases and existing literature, examples of both good and poor practices are presented to support the basic conclusions.

Thesis Supervisor: Dr. David P. Hoult

Acknowledgments

This document is incomplete without recognizing all of those people who made it possible. I would like to thank first and foremost the sponsors of this research, especially those who were kind enough to open their doors and let me poke around for a while. While I cannot name specific individuals due to confidentiality consideration, you know who you are and I greatly appreciate your assistance and great patience.

I would also like to thank the faculty, staff and students of the Lean Aircraft Initiative especially Chrissy, Renata, and Alex, with whom this whole thing got started and Stacey and Maresi who encouraged me to finish it. My advisor, Dave Hoult deserves consideration for putting up with some of my half-baked arguments and forcing them into a somewhat coherent whole.

Finally, I'd like to thank my mother and Dr. Saeed Farokhi for having more confidence in me than I had in myself and encouraging my to come to MIT in the first place. You were right.

Biographical Note

The author has been a research assistant on the Lean Aircraft Initiative through the Massachusetts Institute of Technology's Center for Technology, Policy and Industrial Development for the past three years. He is a native of Ainsworth, Iowa, and has a Bachelor of Science in Aerospace Engineering conferred by the University of Kansas in Lawrence in May of 1992. Mr. Stout worked briefly for the Integrated Network Systems Operations group of Science Applications International Corporation (SAIC) of McLean, Virginia, before attending MIT.

Table of Contents

1. Introduction **7**

1.1 Document Organization Document 7

1.2 Product Development in the Defense Aerospace Industry 7

1.3 The Lean Aircraft Initiative 10

2. Background **13**

2.1 Thesis Topic 13

2.2 Research Topic Selection 14

2.3 Research Process 15

3. Case Histories **18**

3.1 Case A: Electronics System Upgrade Program 18

3.2 Case B: Redesign to Improve Reliability 21

3.3 Case C: Aircraft Block Upgrade and Retrofit 23

3.4 Summary of Case Issues 26

4. Measurement of Product Development **28**

4.1 Two Types of Metrics 28

4.2 Program Metrics 29

4.3 The Nature of "Risk" 35

4.4 Difficulty of Metric Definition & Use 40

5. Elements of Design Decisions **43**

5.1 Basic Elements of Decision Making Process 43

5.2 Appropriate Metrics 44

5.3 Historical Database for Product Development 47

5.4 Decision-Making Process 49

6. Conclusions **54**

6.1 Metrics Not Accurate Predictors of Success 54

6.2 Historical Databases Inaccessible 57

6.3 Decision-Making Process Vital to Good Design 60

7. Next Steps **63**

7.1 Topics for Further Research 63

7.2 Development Issues To Be Addressed 65

7.3 Final Comments 67

List of Figures

Figure 1.1: Importance of Program Phases on Life Cycle Cost 9
Figure 2.1: Distribution of Case Studies 16
Figure 4.1: Metric Framework 30
Figure 4.2: Risk in the Metrics Model 37
Figure 4.3: Sample Risk Magnitude Matrix 39
Figure 4.1: Three Elements of Product Development 44
Figure 4.2: Design Information Flow 52

1. INTRODUCTION

The purpose of this chapter is to provide an introduction to the current product development practices and climate within the defense industry as well as a short history of the Lean Aircraft Initiative, the sponsor of this research. If the reader is already familiar with the defense aerospace industry and the research initiative, this chapter provides little new information.

1.1 Document Organization Document

This document is divided into three sections as listed in the table below.

Section	Chapters
Introduction to Topic	1 - Introduction 2 - Background
Presentation of Research Results	3 - Case Histories 4 - Measurement of Product Development 5 - Elements of Design Decision Making
Conclusions	6 - Conclusions 7 - Next Steps

The first section, chapters one and two, gives the basis for this research as part of the Lean Aircraft Initiative and provides a descriptive introduction to the topic selected. Section two contains the body of the thesis and details the results of the research conducted. The final section is composed of the thesis conclusions drawn from the research given in chapters three, four and five, and suggests directions for additional research.

1.2 Product Development in the Defense Aerospace Industry

As a response to the shrinking defense budgets resulting from the dissolution of the Soviet Union, defense contractors are facing budgetary and competitive pressures that were uncommon during the cold war. These pressures have resulted in rapid changes throughout the industry. Already, numerous corporate mergers, consolidations, and buy-outs have had a profound effect on the make-up of the industry. The remaining defense companies must be increasingly concerned with efficiency and competitiveness to survive in the new harsher environment. In this new climate, it is no longer enough to simply have the most technically capable weapons system available; now such systems must be delivered faster and at a lower cost than ever before. Cost pressures have forced many companies into downsizing which, while providing a short term boost in worker productivity, could have potentially disastrous effects on the long-term health of the industrial base.

To cope with the new higher levels of competition throughout the industry, companies have attempted to integrate many commercial-style practices into their current organizations. Total Quality Management (TQM) techniques, Quality Circles, Just-In-Time inventory systems, Statistical Process Control (SPC) – all have been tried with varying degrees of success by companies within the industry in efforts to increase their efficiencies and competitive positions.

Importance of Product Development Practices

While many of the changes implemented within the industry have been focused on managing and improving the production process, few have addressed the problems in product design and development. In many companies, product development is being performed today in the same manner that it has been done for the past thirty years. Lacking formal standards for product development practices, many companies are finding it difficult to implement wide-scale improvements. While the tools available to the design engineers continue to improve at these companies, the basic process of product development has remained virtually unchanged. Computer-aided design and analysis tools have enabled engineers today to work faster and more efficiently than ever before, but whether these tools have enabled today's designers to work better than engineers in the past is still questionable. The great cultural changes occurring on factory floors have left many engineering departments untouched. This is surprising considering the importance of product development in determining the overall success of the product both in production and in the field.

Product design and development have a major impact on the product image perceived by the customer. While poor manufacturing can easily ruin an otherwise excellent design, even the most gifted personnel are unable to build a quality product from a poor design. In addition to affecting the quality of a product, development also plays a primary role in determining the total cost of the product. Many sources have estimated that the design and development phase of a new program is the primary cost driver within a program. While later acquisition and operation costs may dwarf the amount spent during the design phase of the program, the decisions made during development has been estimated to lock-in anywhere from 80%[1] to 95%[2] of the aircraft's total life-cycle cost. This approximate relationship between the phases of aircraft development, production, and operation and the incurred life cycle cost is summed up in Figure 1.1.

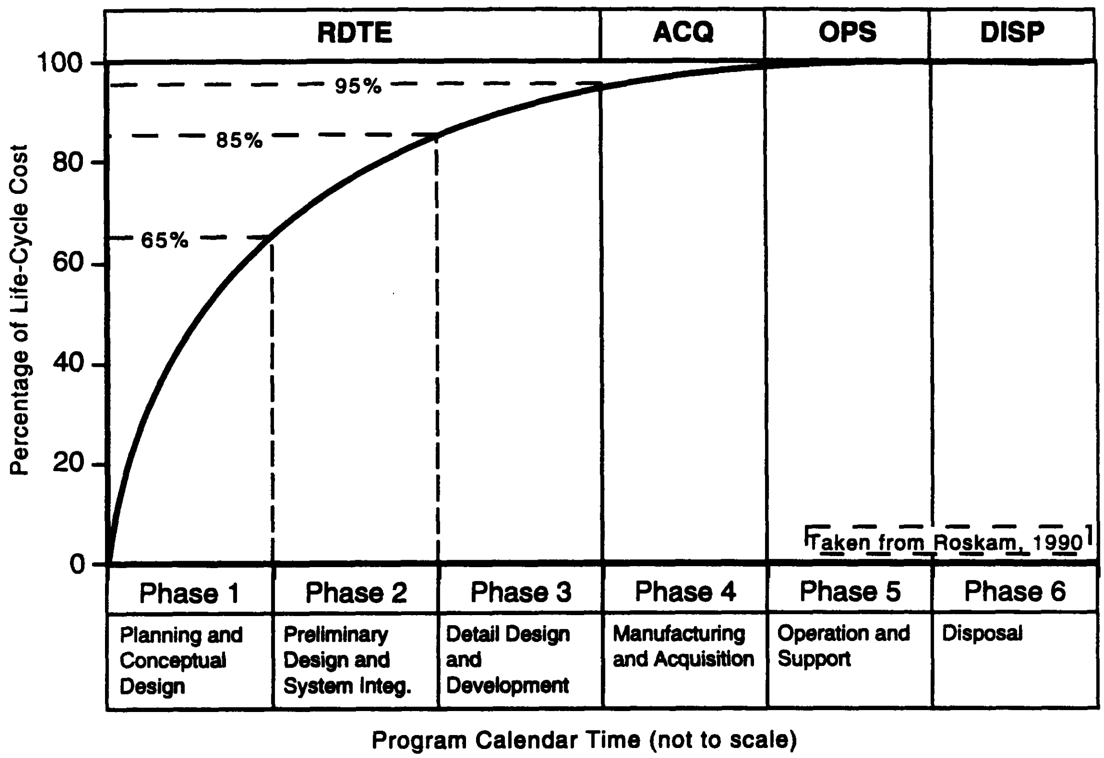

Figure 1.1: Importance of Program Phases on Life Cycle Cost

Movement Toward Integrated Product Development

While there has nearly always been some form of teaming between functional groups during development, in recent years many companies have implemented more formal cross-functional teaming systems. These types of teaming arrangements are

---

1 Fabrycky, Wolter J. and Blanchard, Benjamin S.; *Life Cycle Cost & Economic Analysis*; Prentice-Hall, Englewood Cliffs, New Jersey; ©1991.

2 Roskam, Jan; *Airplane Design, Part VIII: Airplane Cost Estimation: Design, Development, Manufacturing, and Operating*; RAEC, Ottawa, Kansas; ©1990.

often referred to quite loosely as "Integrated Product Development Teams" (IPD Teams). Unfortunately, there are as many definitions of IPD as there are companies. No standard definitions or activities have been set to determine what constitutes an IPD Team and exactly what they are supposed to do.[3]

Within this paper, the terms IPD Team or Integrated Product Team (IPT) will be used to refer to a cross-functional development team designed to integrate up-front many different voices and functions that would normally be addressed in a serial or "over the wall" fashion under the traditional product development paradigm. This definition does not require that the members of such a team be co-located within the company or that members devote 100% of their time to a single project. It does require that so-called downstream functions such as manufacturing and support operations be represented very early in the development cycle (much earlier than they would have traditionally been brought on board) and that the representation of all parties continue throughout the development and production process. While it is understood that many companies have their own distinctive terminology for these principles, the terms IPT and IPD seem to be the most universally understood. The concept of Concurrent Engineering (CE) is assumed to be a part of any IPD team approach and is not treated as a separate issue within this paper. One way of understanding this is that all IPTs, by the definition of this paper, perform some degree of CE, but not all CE must necessarily be done by IPTs. Finally, it must be noted that the practice of production process engineering subsumed within the practice of CE; therefore, the terms IPD and IPPD (Integrated Product and Process Development) should be used interchangeably when referring to the total development process.

1.3 The Lean Aircraft Initiative

The Lean Aircraft Initiative was formed to facilitate the transition to more lean practices within the defense aerospace industry. It originated in the Summer and Fall of 1992 from a "Quick Look" study of the defense aircraft industry to determine whether "Lean" principles, as defined in the book *The Machine That Changed the World*, could be applied to the manufacture of aircraft with the same impressive results as demonstrated in the automotive industry. This book was one output of the research being done by the International Motor Vehicle Program (IMVP) at the Massachusetts Institute of Technology (MIT). The IMVP was a five-year program performing

---

[3] Klein, Janice A.; White Paper — Lean 95-02: "Lean Aircraft Initiative Organization & Human Resources (O&HR) Survey Feedback - Factory Operations; MIT, ©1995.

research on the practices within the world automobile industry. Through this study, the IMVP researchers identified a new paradigm of production separate from both the traditional methods of mass manufacturing and craft manufacturing. This style of production, as exemplified by the Japanese auto maker Toyota, permitted companies to design and build higher quality cars significantly faster and at lower cost than either their craft or mass manufacturing competitors.[4]

The Lean Aircraft Initiative is a three-year program modeled after the IMVP to perform research within the defense aerospace industry. The purpose of this research is to:

- assess and benchmark current industry practices
- identify best practices both within and outside the industry
- facilitate the movement toward lean by the companies

The role of MIT in the Lean Aircraft Initiative consortium is best expressed by the mission statement for the program as given in the original statement of work.

> *"The mission of the Lean Aircraft Initiative is to conduct an organized process of research to define actions leading to a fundamental transition of the defense aircraft industry over the next decade, resulting in substantial improvements in both industry and government operations. Over a three-year period, the program will develop and help implement roadmaps for change, based on systematic and quantitative analyses of current and emerging best lean practices, to achieve significantly greater affordability in the acquisition of existing and new systems, increased efficiency, higher quality, and enhanced technological and economic viability in a competitive international environment."*[5]

From this statement, it is clear that the task facing both MIT and the sponsor organizations is a difficult one. The program goals include not only research results but also implementable strategies to better prepare the participants to deal with a rapidly changing competitive environment.

Lean Aircraft Consortium Membership

Currently, the Lean Aircraft Initiative is sponsored by the United States Air Force and the twenty-two companies listed in Table 1.1. Additional support and

---

4 Womak, Jones, Roos; *The Machine That Changed The World*; MIT Press; ©1990.

5 "Lean Aircraft Initiative Detailed Assessment Planning"; Center for Technology, Policy and Industrial Development, MIT; Delivery Order Contract Number F33615-92-D-5812

direction for the initiative is provided by a number of non-sponsor participants including representatives from the other branches of the U.S. armed forces as well as two of the primary labor unions within the industry, the United Auto Workers (UAW) and the International Association of Machinists (IAM). The cooperation and support of all of these members have been and continue to be crucial for the success of the program.

Table 1.1: Lean Aircraft Initiative Sponsor Companies

AIL Systems, Inc.	Orbital Sciences Corporation
AlliedSignal Aerospace, Inc.	Pratt & Whitney
Boeing Defense and Space Group	Raytheon Company (Raytheon Aircraft Company)
General Electric Aircraft Engines	Rockwell International Corp.
Hughes Aircraft Co.	Sundstrand Aerospace
Lockheed Martin Corporation, Aeronautics Sector	Texas Instruments DSEG
Lockheed Martin Corporation, Electronics & Missiles	Textron Defense Systems
McDonnell Douglas Aerospace	TRW Military Electronics & Avionics Division
Northrop Grumman Corporation	Westinghouse Electronics Systems Group

2. BACKGROUND

This chapter provides the background of the research presented in this thesis. In contrast to the first chapter, which gave more of a general introduction for readers unfamiliar with the defense aircraft industry and the Lean Aircraft Initiative, this chapter focuses specifically on the this thesis, the reasons for its selection, and the method by which the research was done.

2.1 Thesis Topic

The hypothesis of this thesis arose from discussions with a number of industry representatives at the Lean Aircraft Initiative workshop in early 1994 and evolved as the research progressed. In brief, the hypothesis is that the current metrics being used for product development are not alone sufficient for transferring knowledge across development program nor do they act as reliable predictors of successful product development programs. The original goal of this research was to identify metrics for product development that could act as indicators of success as development programs progressed. Through the course of the research, however, the evidence gathered indicated that the difficulties being encountered were not only a problem of improper metrics but one of inadequate communications and a lack of historical records for learning from past developments.

It must be noted that the term "metric" throughout this thesis is being used very generically. While many companies and individuals use this term with a much more restricted definition to include only basic measures of productivity and financial performance, within this thesis metrics encompass any measurements of progress for development programs that provide feedback the engineers and managers. These general metrics include such information as trade study results, technical performance of the product, cost figures and other information broadly comprising the design database that describes the product and process by which it was developed.

2.2 Research Topic Selection

This section provides a brief overview of where this topic originated from with emphasis on the role of the product development survey in defining a potential research area in the application and use of metrics during the product development process.

The Lean Aircraft Initiative Product Development Survey

In the Fall of 1993 into the Spring of 1994, the Lean Aircraft Initiative conducted and analyzed a survey on product development practices within the defense aerospace industry. This survey included many of the largest U.S. defense contractors and covered topics ranging from the degree of design database integration to development teaming practices. The preliminary results and analysis were presented to the initiative sponsors in March of 1994[6]. These results and subsequent discussions identified three broad research areas in the product development area that merited more detailed investigations: the implementation of and attitudes toward integrated product development teams; the use and benefits of cross-functionally integrated design databases; and the role of metrics in driving the design process. Currently, masters-level thesis research is being conducted in all three subject areas[7] with this thesis addressing the issue of metrics for product development. The results of the survey in the area of metrics were very vague. While all companies reported using metrics to monitor the design process, there was no standardization of practices from company to company or even between different divisions within a single company. It was apparent that much up front research would have to be done in understanding and defining the issues before any substantive changes could be implemented.

Description of Research

The research presented in this paper originated from the product development survey and from conversations between the author, thesis advisor, and industry representatives within the initiative. From these conversations, it was indicated that the current metrics being used within product development were inadequate to predict the success or failure of a given design program or to capture the essential lessons learned during development. Since it was felt that many of the engineers and managers working on problematic programs have a good sense for when and how difficulties arise within

---

6 Stout, Todd; "Status of Research Project" presentation; MIT, Cambridge, Massachusetts; ©1994.

7 Browning, Tyson and Dennis, Maresi; both forthcoming. Hoult, David et.al.; "Cost Awareness in Design: The Role of Database Commonalty"; MIT, Cambridge, Massachusetts; ©1994.

those programs, there should be some set of measurements that could be tracked that would indicate when development problems were likely to occur. This research was based on the assumption that such a set of metrics was available and that such a set of predictive metrics would be apparent from a study of past and ongoing development program in which problems had been identified.

2.3 Research Process

To identify such key metrics, the research focused on a series of three case studies to characterize how firms within the industry currently measure and use metrics during the product development process. This section describes why and how the case studies were selected for the research and the nature of the case development.

Why Case Studies?

The case study method was selected primarily because it was felt that a more comprehensive survey of the industry would be impossible under the current time and resource constraints on the research. Focused case studies would allow a deeper understanding of the actual development processes being used by the companies than could be gained using an industry-wide questionnaire. Since each company has their own distinct terminology for many of the processes and activities during development, the generation of a questionnaire form understandable by and applicable to all of the companies is extremely difficult.

The use of case studies allowed the research to focus less on the language being used by the individual development teams and more on the actual practices in question. By selecting a few representative cases, it was possible to gain a deeper understanding of the similarities and differences between development practices in different organizations. This comparison allowed a general picture of the industry practices to be drawn that, hopefully, is more broadly applicable to companies outside the small set of case study participants. While providing a general characterization of current development practices, these cases also provided an understanding of the further research necessary to validate and enhance the applicability of the thesis conclusions industry-wide.

How cases were selected

The cases were chosen to be representative of product development practices and difficulties throughout the industry. While none of the cases identified were designed to be specific examples of best practices within the industry, some of the low-level activities on the individual development teams stood out as at least the best within the case study group. Without a more comprehensive, in-depth look at a larger sampling of the sponsor companies however, there is no way to truly identify best practices for the specific topics discussed in this paper.

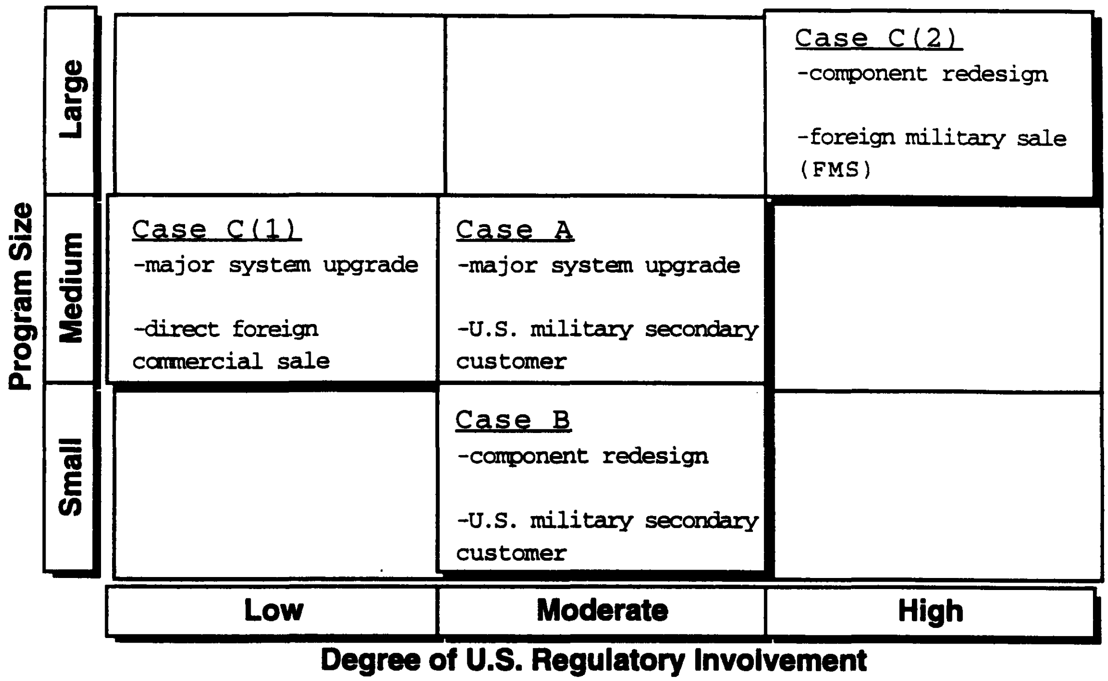

Figure 2.1: Distribution of Case Studies

The cases covered a wide range of practices and program types. Figure 2.1 presents a categorization of the cases investigated by the size of the program and the degree of involvement by the U.S. government. Two of the three cases involved the development of products deliverable to the U.S. military while the third case was for a military sale to a foreign government. All of the cases involved modifications or upgrades to existing systems and platforms. No entirely new aircraft developments were included in the case study sample. While the decision to use a wide range of case types limits the depth of the evidence available, it does allow for broader generalities to be drawn from the commonalties between the three cases.

Limitations of Case Studies

The greatest problem with basing much of the evidence in this thesis primarily on case studies is that there is no guarantee that the conclusions gathered from the cases are representative of the industry as a whole. In the most extreme case, this would mean that the results applied only directly to those companies that participated in the case work. We have attempted to avoid that result by selecting cases that seem to be representative of the entire Lean Aircraft Initiative sponsor group if not of the entire industry. The cases used and preliminary observations were presented to the Lean Aircraft Initiate focus group on product development during the summer of 1994 to ensure that the evidence was indicative of problems and approaches being used in the industry as a whole; however, without additional research to validate the thesis conclusions this criticism will remain.

Case Histories & Program Descriptions

As previously indicated, all three of the cases studied represent examples of modifications or upgrades to existing hardware. Each of the products being designed must not only fulfill new technical requirements, they must also operate with a current hardware platform. The three cases presented in this paper represent three distinct approaches to product development in the current defense environment. While they all were performed under the umbrella of current defense acquisition rules, the development methods and tools used by each program vary considerably. These cases provide a cross section of modern product development in a defense environment.

3. CASE HISTORIES

This section presents brief case histories for each of the three primary cases discussed in this paper. Although unattributed within the footnotes to this document, most of the information in this chapter was taken from discussions and personal interviews with representatives from the case study participants. To maintain the anonymity of the companies involved with this study, the three cases will be referred to as cases “A”, “B”, and “C”.

3.1 Case A: Electronics System Upgrade Program

The first case (Program A) is an example of a system upgrade being provided to a prime contractor for inclusion in new aircraft being produced. While independent research studies had performed for about a decade, the development portion of Program A officially began within the last five years. The design work occurred over an approximately four year timeframe with full-scale development ending about one year ago. At the time of the interviews for this case, the program was entering the production phase, having just completed a production ramp-up.

After being awarded the contract for development of the upgrade system, Company A was informed by their customer that the funding profile originally negotiated would be altered due to the shrinking defense budget. To avoid losing the contract, Company A agreed to the new terms. These terms, while reducing the level of funding considerably, maintained the original technical requirements and product delivery schedule. This required Company A to cut the costs of the program while still maintaining the same development pace and design goals. During this cost reduction process, the decision was made to reduce the number of engineering prototype units available for the hardware and test equipment engineers during the development.

The first engineering prototypes were delivered approximately two years into the development program. A series of four more development units and two pre-production units were produced in the months to follow. This prototyping program was unique for Company A in that it was the first program where the engineering units were produced on the same production line where they would eventually be made.

Traditionally, engineering prototypes had been built by the designers in their own fabrication facility adjacent to the engineering sector of the company. While these early prototypes were manufactured primarily by production personnel, they consisted of a mixture of production and non-production parts and processes. This fact limited their use as a process de-bugging tool for the manufacturing people. Although they were able to provide more input to the designers than if they were not involved with the prototype assembly at all, there was no attempt to simulate the actual manufacturing conditions and constrains during the manufacture of the engineering prototypes.

These "hybrid" prototype units also posed problems for the engineers designing the test equipment. Rather than waiting for the design to be completed, the test station and test process design were done concurrent with hardware development. However, shortcuts in the development phase bypassed the verification of the acceptance requirements. Validation was therefore not done until the production phase of the program. This caused late and premature releases of the acceptance requirements for production.[8]

The concurrent nature of this design process, combined with the reduced number of development units available created a demand for the engineering prototypes that far outstripped the available supply. Since the de-scoped budget did not provide enough development units for each functional group to have their own, after each new development unit release a power struggle ensued between the functional engineering groups that needed the hardware to continue their design work. Primarily, the struggle was between the product engineers, who needed the units to further refine the hardware being developed, the test engineers who were designing and building the test equipment to be used in production, and the software development engineers who needed the units to work on the operational-mode and built-in-test software.

Ironically, the software development group, considered by the company to be the "riskier" development, was exempted from many of the problems faced by the hardware developers. Because software development had traditionally posed a greater risk for this company than hardware development, more engineering prototypes were provided to them than the hardware designers. This resulted both in a smoother software development program than expected and in an increased degree of resentment

---

[8] Test engineer at company A.

(For purposes of confidentiality, all attributions and citations in this section shall remain anonymous) from the hardware development team. The future effects of this slightly altered attitude toward the software development people have yet to be seen.

As a whole, the development program was viewed by the customer as a resounding success. Even in the face of the de-scoping that had occurred, the program was able to deliver a design that satisfied the customer requirements while still being on schedule. Although many of the design engineers were concerned about the "shortcuts" that had been taken to achieve this success, the program proceeded into manufacturing being hailed by the customer as one of the smoothest and best development projects they had ever been involved with.

As manufacturing began, however, the concerns of the engineers began to materialize. Nearly all of the production units failed their operational tests and were unfit for delivery to the customer. At first, there was suspicion that faulty test equipment was to blame for many of the errors, but upon further investigation this was found to not be the case. Whereas the engineering prototypes had performed well during tests at ambient conditions, when subjected to harsher testing over the entire operational range the production units exhibited a nearly 100% failure rate. Due to the lack of available prototype hardware, thermal testing across the operational range had not been done during the development process, and now that omission was becoming a problem during manufacturing.

Although nearly all of the production units failed during first pass testing, most of them could be reworked enough to pass on their second or third try. Working units could still be delivered to the customer nearly on the original schedule, but the rework content added significantly to the cost of each one delivered. Additionally, nearly every unit being delivered failed to satisfy all of the contracted performance criteria. These units were shipped to the customer with waivers for each deficiency discovered. The waiver process, itself, added significantly to the costs of the production units and also increased the future liability that the company would be subject to if the delivered units were incapable of meeting the ultimate customers' expectations.

In another ironic twist, the manufacturability problems discovered after production began were seen by the customers as failures in the production phase of the program, not as design deficiencies. While the development was hailed as a model program, it had succeeded in creating a nearly unproducible design.

The redesign effort addressed many of the producibility problems of the original design. While production continued on the faulty design to meet delivery schedules, over one-third of the major components had to be altered in the course of the redesign work.[9] In some cases, the changes implemented might have to be integrated into the upgrades already produced. This requires that additional time and expense be incurred to provide the necessary hardware, training and personnel to perform the retrofits on the previously installed upgrades. In addition to the configuration management problem that this creates for both Company A and their customers, it also introduces a serious liability concern for the company. If the deficiencies of the currently shipping systems are found to impact the future performance of the system, Company A could be held liable for bringing all of the installed systems up to date at their own expense. Rather than saving money, the "de-scoping" of the project has already resulted in costs exceeding the initial saving and could potentially result in even higher costs to the company in the future.

3.2 Case B: Redesign to Improve Reliability

Case B is an example of a focused redesign effort to fix a problem with a system currently in use. This design addresses only a small part of the system in question and is directed toward providing a solution the customers' problems rather than performance enhancements. As such, the scope of this development effort is much smaller than that discussed in Case A; however, it too exhibits many of the same characteristic problems.

The system in question was introduced into service over a decade ago and has been experiencing seemingly random failures over nearly its entire lifetime. While many of the failures of this system in the field were attributed to a single part, nearly a dozen prior redesign efforts had failed to adequately solve the problem. Although all of the redesigns could see what the problem was, none of them were able to accurately identify the root cause of the failures being encountered. They were primarily concerned with providing fixes that would satisfy the customer rather than rooting out and eliminating the cause of the failures entirely. The closest that any of these redesigns came to finding a bandage large enough to cover the problem was probably the redesign effort that immediately preceded the one studied in this case.

This prior redesign will be referred to as the "first" redesign. While it was by no means the first redesign attempted, it was the first that any of the people interviewed for this case were actively associated with. This first redesign focused on providing a solution to the customer as quickly and inexpensively as possible. The designers used failure histories to isolate the general problem and proposed a solution that seemed

9 Test engineer at company A.

appropriate. Unfortunately, the scope of their work did not include a full analysis of the recommended solution. When this analysis was later performed it showed that while the proposed solution would alleviate some of the problems being faced, it probably would not eliminate the failures entirely. While the "solution" had attacked the symptoms of the problem, it did not address the root problem causing the failures.

One of the recommendations made prior to the first redesign team that might have been successful at addressing the cause of the problem was never implemented. A specification for maintainability required the part to be "field serviceable". This means that instead of replacing the part with a new one when they fail or become damaged, the maintenance personnel in the field must be able to repair the part with available materials. Although this field repair almost never occurs in practice due to the time and difficulty of such a repair, the specification still dictated that the ability to do so must be maintained. Since the proposed change would have adversely impacted the maintainability of the part and would have required changing the specification, it was vetoed by company management before every being presented to the customer for consideration.[10]

Although there is no way to know if the rejected design would have solved the problem, it is known that the solution implemented by the first redesign team did not. At first, the solution looked promising. The number of failures the customer experienced during acceptance testing and the first few weeks of operation fell to near zero. As the systems went into operational use, however, the same types of failures began to reappear. The first redesign had succeeded in delaying the problem, but not in eliminating it.

Following the Persian Gulf War in 1991, it became evident that these persistent failures being encountered were severely hampering the reliability of the entire aircraft. A greater urgency was attached to finding a real solution to the problem and Company B was again charged with redesigning the portion of the system that was causing the most problems. This time, however, a greater emphasis was placed on determining the root cause of the problem and removing it rather than just providing yet another patch that would relieve the symptoms but fail to cure the disease. With this mandate, Company B put together a team of engineers to finally solve the problem.

To achieve their goal of providing a final solution to this persistent problem, Company B implemented a focused team approach toward finding and eliminating the

---

[10] Supplier representative for company B.

cause of the failures. The engineer originally chosen to lead the team was given a free hand to manage the composition and activities of the analysis and redesign team.

Even with a new commitment to provide a complete solution, the team assembled to complete the analysis and redesign to solve the problem still ran up against a number of constraints. The primary barriers they encountered included a poorly defined program goal and schedule constraints. Originally, the redesign effort was limited to the five primary connection interfaces where the majority of the failures occurred. Secondary connection interfaces on the same item that exhibited very low failure rates were not included in the original scope of the redesign.

After the analysis of the faulty interface had begun, the customer indicated that a more complete analysis that included the secondary connectors would be desirable. To please the customer, the scope of the project was expanded to include all of the connections on the problem hardware. The project continued with this broader scope for a few months until it became evident that it would be impossible to meet the delivery deadline for the redesigned hardware unless the project was more focused. While the team recognized that a full analysis of the entire article was still important, Company B chose to refocus the team's effort back onto the original goal of redesigning the primary connectors.

The redesign team worked toward meeting the strict delivery schedule even though this meant sacrificing an opportunity to complete a full analysis on the problematic hardware. While they now understand the current failure mechanism better than ever before, they are unable to predict whether failures in the secondary connectors will now become the primary failure modes, nor can they determine how often these new failures will occur over the life of the product. They hope that the faults identified in the focused analysis are the only failure drivers, but until the new solution has actually been installed and used in the field, the redesign team has no way of knowing if their efforts were successful.

3.3 Case C: Aircraft Block Upgrade and Retrofit

The third case, Case C, while also examining an upgrade of an existing system, contains some factors that make it unique from the other two. This case is the only one of the three presented whose ultimate customer is not a branch of the United States' military. Additionally, one portion of this program is being performed as a direct commercial sale outside the umbrella of United States' federal acquisition regulations. The remainder of the program is being handled as a foreign military sale (FMS) and is subject to the same acquisition regulations as direct purchases by the United States' military. The dual nature of this development effort allows for comparisons between the development process under the current system and the same process with more relaxed regulations.

The direct commercial portion of Case C concerns the development of upgrade hardware for systems that the customer already owns. The overall development scope for the commercial portion of the program is the smaller of the two parts, for most of the common elements it is being done first. Whereas it is usual for an upgrade to bring existing systems into parity with newly acquired ones, these system upgrades are being developed for currently deployed aircraft before being produced as part of a new aircraft purchase. Therefore, many of the non-recurring costs will be either split between the two programs or borne entirely by the commercial contract. The burden of this cost sharing arrangement has fallen mostly onto the fixed-price commercial contract.

Unlike the FMS portion of the contract, the commercial development and production program is being performed on a completely fixed-price basis to the customer. As such, cost overruns in this program will be the sole responsibility of Company C, and cost savings will be directly related to their profits. This more commercial-style business arrangement has forced Company C to streamline some of their business practices and has greatly increased the pressure on the company to control their development costs. One manager of the program estimated that in his portion of the design, the commercial development contract had potential savings of over 20% when compared to the FMS development costs.[11] Even with these lower costs, however, the price paid by the customer for this upgrade system was not significantly lower than if it had been handled as an FMS. Because the commercial contract placed all of the risk squarely on Company C, they demanded that their rewards be commensurate. For the contract to be acceptable, Company C's potential profit had to be higher on the commercial program than on the FMS one. The net effect of handling the development as a commercial sale was to eliminate the middle man of the U.S. government.

To purchase new aircraft containing the upgraded systems, however, the customer was required to funnel the transaction through the channel of an FMS. To company C, this FMS appears just like a sale to a U.S. military customer. While the FMS portion of the program benefited from the advance development work being done on the commercial contract, they had many additional costs associated with satisfying

---

[11] Project team leader, Company C.

U.S. procurement regulations. Although the systems being produced are sold to a foreign customer, Company C still must fulfill all of the requirements of a domestic military purchase. These include documentation requirements much broader than those that the customer requested on the commercial buy and a higher degree of oversight by government inspectors.

In some ways, however, Company C does benefit by using the FMS system. Foremost among these is that the company is subjected to much less risk on the FMS sale than on the commercial one. Although actual delivery contract terms had yet to be negotiated, the existing long-lead contract covered the development work similar to a cost-plus contract. While there were definite price goals negotiated with the customer, cost overruns in the development program were not borne fully by Company C as was the case for the strictly commercial contract. Additionally, since the majority of Company C's other contracts are either FMS or direct U.S. military sales, the entire company infrastructure is focused toward satisfying the acquisition requirements. In fact, even though they were not requested by their commercial customer, Company C satisfied many of the requirements of an FMS simply because they had always done it that way before.[12] While these activities may not have been optimal in a purely commercial development environment, for Company C they were just business as usual.

The fact that a large portion of the upgrade hardware being developed had to be installed on both new and existing aircraft allowed for coordination between both development programs to reduce the overall cost of the systems. This coordination is evident from the organizational structure used by the development teams. Instead of employing two completely separate development team structures, Company C chose to form two overlapping teams. Where the products were to be similar between the two programs, a single sub-team was formed that had dual responsibilities to each of the independent team leaders. This allowed resources to be shared between the two programs and avoided many of the turf battles that would otherwise have occurred between two such similar programs. Additionally, these links allowed a mechanism for information transfer between the two programs that would normally not have existed. Because of the use of personnel common between both of the development programs, problems encountered by one program were quickly made known to the other program, and many potential pitfalls were avoided as a result.

---

[12] Program manager, company C.

The joined development team structure was not entirely beneficial. This structure added a further complication to the cost distribution on both programs. While the ultimate customer was the same for both programs, the contracting mechanisms were separate. Thus, the costs of each program had to be maintained independently. To achieve this using the linked development teams, a system of dividing the work between the two programs was devised. Although Company C attempted to fairly split the costs incurred on activities beneficial to both programs, often the direct commercial contract was forced to assume a larger portion of these shared costs. Since the commercial program was subject to an earlier delivery date, much of the common work was performed prior to the need date for the FMS program. This shift in the development schedule created an additional marginal cost to the FMS development items that was extremely difficult to estimate. To avoid the appearance of impropriety, the entire development cost of many common items was borne by the commercial contract. Since the cost savings this generated in the FMS program were not passed directly on to the company as they were in the commercial sale, this inability to accurately distribute the development costs between the two programs resulted in a slightly reduced profit level for the company as a whole.

3.4 Summary of Case Issues

In all three cases discussed, the storage and dissemination of design knowledge play a major role in determining the ease of the current development activities. A better capturing and communication of the lessons learned by past design teams would have benefited the current design teams in both Case A and Case B while communication between disparate design teams with radically different motivations and goals is of primary concern to the two teams in Company C. All three of these design teams as well as other engineers within the consortium recognize the importance of inter-team and inter-generational communication of design knowledge.

Traditionally, the communication of lessons learned has been relegated to the functional design organizations. In the past, long design cycle times, highly functional organizational structures, and low employee turn over have contributed to an atmosphere pseudo-apprenticeship within many engineering departments. With the modern movement at many companies toward shorter design cycles, flattened team-based organizations, and early retirement incentives, there is great concern throughout the industry that much of the half-century of accumulated organizational knowledge is in danger of being lost. While the passage of "tribal knowledge" from generation to generation within functional organizations has been adequate in the past, the changing nature of aerospace product development may necessitate a more proactive approach toward the gathering and dissemination of knowledge throughout the organization.

4. MEASUREMENT OF PRODUCT DEVELOPMENT

In all of the cases studied and from discussions with the research sponsors, the measurement of product development performance is considered a critical key to the success of any development activity. These measurements generally are in the form of metrics that are tracked by the program or company throughout the course of the development activity. Typically, the metrics used are either program-level or process-level indicators of one or more of the three basic measures: design cost, design time (schedule), and technical performance. To make any assessment of the current metrics, we must understand how they relate to these three primary metrics. This requires a common, basic framework for examining how the basic measurements relate to each other and how they relate to program risk factors.

4.1 Two Types of Metrics

One of the principal problems with many current metrics is that they often seem rather schizophrenic to the user. This problem lies in the fact that there are really two basic types of metrics in use, and that the distinction between these two types is often vague. The first type of metrics is program metrics. Program metrics are measurements that allow companies or design teams to follow the progress of a single product development program. They are typically compared to expected values based on historical data or prior performance. These metrics are generally reactive in nature and only indicate difficulties after problems have already occurred. Program metrics, by their very nature, do not facilitate process comparisons or improvement over past performance well. They are mostly used as up font planning tools and as status indications for individual programs.

The other primary measurement categorization is process metrics. These metrics provide the bases for comparison between programs, companies, and individual workers and are often used to make benchmark comparisons against the best practices within the industry. Within a specific company, these metrics provide a reference by which the engineers, workers, and managers can gauge individual progress or proficiency. They are most useful in evaluating improvement over time and often have idealized or unachievable goals such as zero defects in a production operation. While these process metrics allow companies to view the basic processes of product development in standard quantifiable terms and to compare their current practices against those of past development teams or other firms, they offer little insight into the day-to-day progress of individual development programs. While the performance measured by the process metrics might have the most direct influence upon whether future product development programs will succeed or not, it is the more highly visible tracking metrics by which the actual success of the program is defined.

A useful metaphor to visualize the difference between program and process metrics is to imagine the construction of a house. The program metrics can be thought of as measurements of the construction timetable: site selection, laying of the foundation, framing and finish carpentry work. The program metrics track the progress toward the final goal of a perfect completed house. Measurements along this ideal timeframe are neither good nor bad in their own right, but only take on such qualities when compared to the overall program plan of what each measurement should be. The process metrics, however, relate to the quality of the products and processes that go into the construction of the house: the skill of the carpenters, the grade of the lumber, the strength and uniformity of the bricks and mortar in the walls. While it is possible to build a house exactly to the specification of the blueprint using inferior materials and workmanship and it is likewise possible to use the highest quality materials and laborers and still not conform to the specifications; to obtain a high quality final product, both the program plan and the quality of work must be of high quality.

4.2 Program Metrics

In each of the cases, metrics were used to provide feedback as the development process proceeded. Nearly all of the metrics used in the cases were simple combinations of the three primary measurements: cost, schedule and technical performance. These three primary measurements can be thought of as axes in a three dimensional coordinate space as indicated in Figure 4.1. Within this space, the planned performance and actual performance can each be plotted as a track from the origin to a point representing the completed project, the point (1,1,1) in the figure.

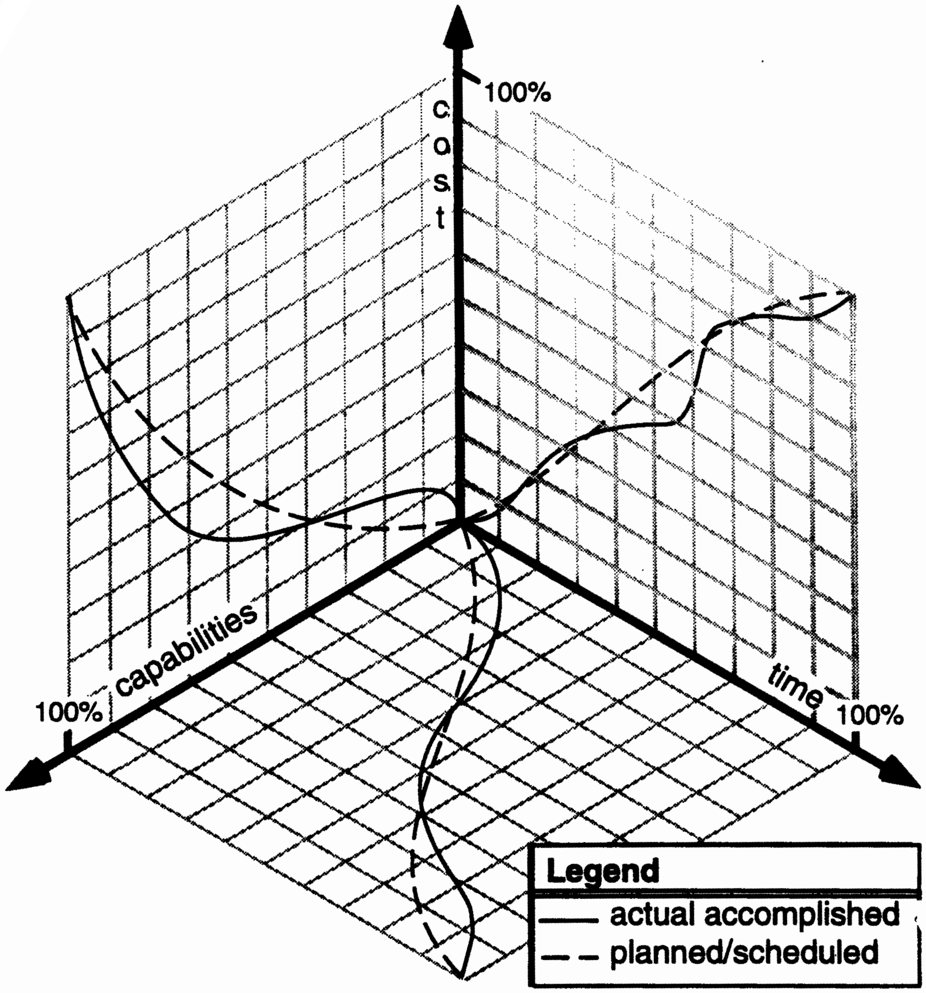

Figure 4.1: Metric Framework

If these plan and actual tracks are projected onto the three planes formed by the three primary metrics, the result is three plots familiar to the engineers and program management in Company C. The projection onto the cost-time plane results in a plot of the program budget versus the actual expenditures of the program. Likewise, the projection onto the technical capabilities–time plane results in a plot of the planned and actual completion of program activities, and the projection onto the capabilities–cost plane results in a plot of the budgeted and actual costs of achieving the program technical objectives.

Basic Measurements and Examples from Cases

The final targets for the primary metrics (the levels defined as 100% in the model) are generally projected at the start of the program and specified in the development program contract. These primary metrics can then be tracked by comparing the current state of the development to overall program plans for each of them. For example, schedule performance consists of the actual program pace compared to the pre-planned schedule laid out at the start of the program.

Program Schedule

Of the three primary metrics in the cases studied, the significance of the time schedule outweighed both cost and technical capabilities. While the technical capabilities could always be rescoped and costs renegotiated, in each of the cases studied the schedule could not be broken. Even in Case A, where normal development activities had been pushed back into the pre-production phase of the program, none of the people involved with the program would accede to calling this a schedule slippage. The managers and chief engineers of the program were adamant in maintaining that the original program schedule was being maintained although this was only achieved by shipping products that did not satisfy the original technical performance requirements.

The program schedules in this industry are typically laid out on a variation of the standard system of milestones and reviews. Individual team leaders in conjunction with their teams and each other determine the activities and steps that are necessary to achieve the individual team objectives. Using best guesses and historical data from prior development efforts, each of these activities is assigned a time duration required for completion. The set of contiguous activities that places the greatest constraint upon the total time required is identified as the "critical path". This critical path of activities and their associated durations dictate the overall length of time required to perform the complete set of tasks. This ordering of activities is repeated until all activities required to produce the finished product are included. In most cases, this entire set of activities is then reverse scheduled from the contracted delivery date or other fixed program milestone. After this master schedule is set, the actual time required to complete the tasks set forth in the program plan can be tracked against the projected schedule. The accuracy of this comparison provides a basic measurement of the performance of the development team.

One of the primary benefits of maintaining a schedule of this kind comes not from the ability to retroactively track the progress of the development against the planned schedule, but from the planning of the schedule itself. By forcing the development teams to examine in detail the tasks required to achieve the proposed development effort, teams are able to gain insight into the magnitude of the program and to identify linkages between program elements that could go unnoticed without a detailed program schedule.

To realize these benefits, however, task scheduling must be performed in close coordination with the actual development team members. Through the use of critical path or similar methods, links between disparate portions of development programs become evident when viewed in a coordinated cross-functional development environment. For highly compartmentalized companies or extremely large development programs, these linkages are often difficult to observe without some form of coordinated scheduling effort. In Case C, for example, program schedules were laid out to an extremely detailed level, but there was no standard process or mechanism in place to coordinate these schedules at anything greater than a milestone level. While the company maintained a master schedule for the entire development activity, this schedule focused primarily upon major program events and did not have enough detail to allow identification of low-level conflicts between individual design teams. For uncoupled designs where components can be developed separately with little interaction between the separate development teams this type of approach might be acceptable, but in situations such as in Case C where each component-level team depends upon nearly continuous interactions with teams working on other elements of the system, the lack of integrated and well-maintained development schedules can pose a serious coordination problem.

An extreme case of this lack of schedule coordination can also be found in Company C that is similar to the problems observed with separate project teams. A major component required for the development program at Company C was being produced for the customer by a third party. Originally, there were no provisions for direct interaction between Company C and the outside supplier. Both parties relied upon the customer to coordinate the activities between the two. This lack of direct contact led to a serious scheduling problem. While the customer had coordinated the two companies' product delivery schedules to accommodate their own desired delivery dates for the final product, they had failed to adequately account for lead time that Company C needed to integrate the component into the rest of the system being developed. To resolve the matter, Company C and the third party negotiated with each other directly to achieve the necessary schedule coordination that would allow the production to meet their delivery goals. This example is analogous to the arrangement within Company C between the individual component development teams. While the close proximity of the teams allows for better communications than the company had with the third party, the lack of up front coordination still results in the necessity for "renegotiations" of delivery schedules between the teams on down the line.

Product Technical Capabilities

The second primary measurement is the technical capabilities of the product. While there is usually no single metric to measure the overall technical level of the entire product, all of the program examined did track the individual technical performance characteristics in some manner. These performance measurements often take the form of the degree of satisfaction of the technical requirements of the contract or a demonstration of progress toward those requirements. In general, the technical capabilities are specified as performance benchmarks that must be satisfied for customer acceptance of the product. While having such rigid approval criteria helps to define the acceptable design space for the products being developed, it relies heavily upon the contract author's ability to accurately translate customer needs into specific contract requirements. When this translation is incomplete it is possible for the resulting product to satisfactory all of the contract requirements yet still fail to adequately meet the original needs of the customers. As with the measurement of the program schedule, the most difficult aspect is not the measurement of the actual product performance, but rather it is the development of the yardstick by which the performance will be measured.

In the cases studied, nearly all of the design engineers interviewed treated the overall contract requirements as rigidly fixed. While trades between technical performance and either cost or schedule could be performed at the detailed design level, solutions that adversely affected the product's performance to the contract requirements were discounted, even if they had effects, such as cost reduction, that were desirable from a non-performance standpoint. More than one engineer expressed the feeling that low-cost solutions were good only as long as they did not hinder the overall performance of the system.

The focus on satisfying rigid contract requirements creates constraints on the acceptable design space. When the specifications are too rigid or are defined to a too highly detailed level, the ability to trade levels of performance for other factors, such as cost, is severely hampered. The leader of one design group was quite surprised at the suggestion that the technical requirements of the product should be treated as a variable when doing overall design trades. He remarked that while elements of the program cost, schedule and risk were all actively traded with each other, the product performance, as specified in the contract, was treated as a constant simply because it was so difficult to get the wording of the contract changed. Even if minor degradations in performance could result in large savings to the company or the customer, they normally were not pursued since it was easier to meet the requirement and let the program costs increase. [13]

Although the overall performance targets of the product were considered fixed during development, these technical capabilities were still secondary to the program schedule when the two conflicted. If the decision was between shipping products on time that did not meet the performance requirements and delaying shipments of those products to bring the performance in line with the requirements, the former would be preferred nearly every time. As previously mentioned in Case A, for example, financial constraints during the product design caused the development of a product that was unable to meet of the specified technical requirements. Instead of delaying the shipment of the problem goods, however, products were shipped to the customer with waivers for the performance deficiencies. This waiver process meant that Company A was still financially responsible for making the deficient products comply with the original requirements, but that they would be allowed to ship the units on time and avoid the penalties for late delivery. Rather than reduce the technical capabilities expected from the product, future financial risk was traded for the ability to satisfy the original schedule.

Program Cost

Finally, the third primary measurement is program cost. While it may initially appear that cost should be the primary driver during product development in a downsizing defense environment, in the cases studied, this did not seem to be the case. Although low costs were viewed as important to the overall success of the programs, costs overruns were not viewed with the same anxiety as schedule slippages or performance degradations. This attitude was most pervasive in the cost-plus contracts, but was also evident from fixed-price contracts because of risk sharing arrangements with the customer. In general, costs were viewed as incidental to the product development process and were not emphasized as heavily as either program schedule or product technical capabilities. Costs generally were treated as the results of a difficulty with one of the other primary metrics. Schedule slippages were to be avoided because they increased the program cost, for example. When faced with a choice between incurring the direct cost associated with slipping the schedule or incurring the risk of a

---

[13] Program manager, Company C; June 1994.

much larger downstream cost, the case evidence indicates that companies are more likely to hold the schedule and take their chances with the large downstream cost.

Program costs were viewed as inflexible only in the portion of Case C performed under a commercial-style, completely fixed price contract. Under this contract, a final price was determined between Company C and the buyers up front with no provisions for risk sharing if the program ran into unforeseen difficulties. This meant that any potential cost overruns would be deducted directly from the profit that Company C received. On this program, the costs were viewed as the most critical aspect of the program and placed noticeable constraints on the development team. While this added pressure did help the design team to focus on cost reductions and drive down the cost of the product significantly, when compared to a similar product delivered through a standard foreign military sale arrangement, the price to the customer was reduced only slightly due to the higher profit margin on the commercial contract. It is important to note that costs were driven down only when the profit incentive was high enough to justify the additional risk of taking on a fixed-price development.

An additional complicating factor for program managers trying to reduce the project costs was a general inability to characterize the true costs of product development activities. While it may be helpful in some cases to track the program costs in a gross manner, generally such a practice is not precise enough to pinpoint the cause of budget problem until well after they occur. This problem is due primarily to the fact that many of the costs assigned to development efforts are characterized as indirect costs. In extreme cases, these indirect expenses can be so large as to actually overwhelm the true costs of performing the design. The actual costs of doing product development work are lost among other costs and charges to the program. One striking example of that can be found in the statement by a development team manager from Company C who, when asked whether he thought that having financial authority over the design team's budget was important, replied that it really didn't matter because he could only directly control about a third of the money anyway — the rest was just baggage that came along with the direct costs.[14]

4.3 The Nature of "Risk"

The single factor that ties together all of the development metrics is risk. Risk can be represented in the model of Figure 4.1 by comparing the actual performance to the planned development schedule of activities. The current program risk at any given

[14] Design team leader, Company C; June 1994.

time can be thought of as simply the ratio of the amount of time or effort behind schedule to the amount of time or effort remaining on the original schedule. To illustrate this, a sample projection of the development plan and actual development activities onto the time-technical capabilities plane is presented in Figure 4.2. In this sample plot, the current risk level can be calculated as the difference between length of line segments A and B divided by the length of B. The singularity produced when B approaches zero can be eliminated by taking the arctangent of the ratio and normalizing this result to 1.0 at the extreme points of ±90°. When the resulting fraction is multiplied by the consequences of the current deficiency, δ, an approximation of risk is obtained. In equation form:

$$\text{Current Risk} = \delta \cdot \frac{\tan^{-1}\left(\frac{A}{B} - 1\right)}{\pi/2}$$

(where δ, A and B are defined in Figure 4.2)

While this representation of risk does provide a simple measurement for risk using easily measurable metrics, it does little to address the important questions of future risk or the inherent riskiness of the program as a whole. This future risk can be thought of as the probabilistic additional costs (either in time, money, or performance deficiencies) incurred in the program due to uncertainty of the decisions being made. In a basic form, risk can be represented as the consequential cost of failing to satisfy one of the metrics multiplied by the probability of that failure. For elements of the program where the probability of failure is zero, the risk is likewise zero; for elements that are doomed to fail, the risk is equal to the full value of the consequences of the failure. Ideally, the sum of all of the risk factors is driven toward zero where the success of the program is assured.

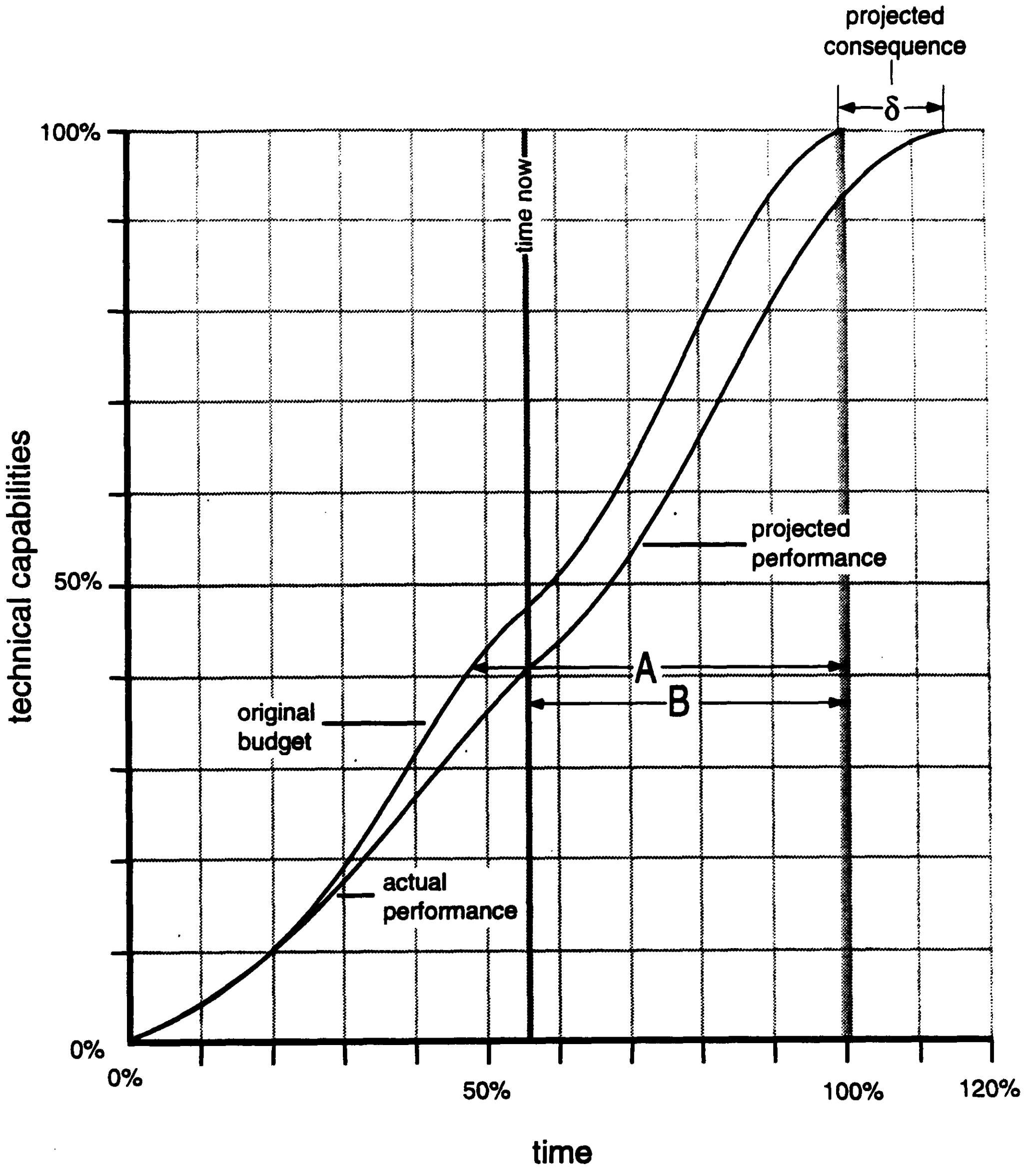

Figure 4.2: Risk in the Metrics Model

In practical usage, however, this probabilistic measurement of risk is almost never used. While the consequences of failure can often be approximated through some form of fault tree or failure mode analysis, the probability of failure is nearly impossible to foresee in complex organizational systems. Risk measurements, therefore, are often supplanted by risk management programs relying on the subjective assessments of the program personnel involved.

The risk management employed by Company C provides a good example of current industry practices. Instead of analytically evaluating the consequences and probability of failure on a continuous scale, their risk management process defines five broad subjective levels to each one. An overall risk factor of either low, moderate or high risk is determined by looking up both the probability and consequence on a matrix similar to the one reproduced in Figure 4.3. This overall risk factor can then be compared with other risks within the program or "rolled up" into an overall risk rating for the program as a whole. Once the risk factors have been determined, the team members can then choose how to address these risks. The four risk management techniques identified in Company C's process manual are: Avoidance, Abatement, Control, and Acceptance. After selecting the management strategy to be applied to each risk factor, additional process-level metrics are selected to be used to monitor the health of the program and insure that the moderate and high risk items have been adequately addressed. In general, this first level risk management is all that is performed during any single risk management. While the risks may be assessed repeatedly for long programs, normally the first level risk management strategies are not reevaluated as potential sources of additional risk.

Likelihood	Consequence / Definitions	Negligible	Minor	Moderate	Serious	Critical
		Has insignificant effect on program	Small increase in cost/schedule; tech. requirements still met	Moderate increase in cost/schedule; tech. requirements still met	Major increase in cost/schedule; tech requirements still met	Jeopardizes program ability to meet tech. requirements
Frequent (91–100%)	Likely to occur often throughout program lifetime	MODERATE	MODERATE	HIGH	HIGH	HIGH
Probable (61–90%)	Likely to occur several times during program lifetime	MODERATE	MODERATE	MODERATE	HIGH	HIGH
Occasional (41–60%)	Likely to occur sometime during program lifetime	LOW	MODERATE	MODERATE	MODERATE	HIGH
Remote (11–40%)	Unlikely to occur but possible	LOW	LOW	MODERATE	MODERATE	MODERATE
Improbable (0–10%)	So unlikely that non-occurrence can be assumed	LOW	LOW	LOW	MODERATE	MODERATE

Figure 4.3: Sample Risk Magnitude Matrix

4.4 Difficulty of Metric Definition & Use

Although each case investigated employed product development metrics to some extent, often engineers expressed difficulty with many of the metrics they were asked to track and had even substituted basic rules of thumb in places where they felt the current metrics were lacking. Other than the three primary metrics of schedule, budget, and technical capabilities, many of these engineers had difficulty explaining the use of some of the composite metrics they were tracking. Most of these difficulties stemmed from basic misunderstandings about the nature of the metrics and how they should best be applied.

Nature of Defense Aerospace Development Changing

A contributing factor to the confusion over the definition and use of product development metrics is the rapidly changing nature of the environment in which the metrics are to be applied. As discussed in Chapter 5, the proper use of metrics in the design decision making process relies upon the availability of adequate historical data to allow for interpretation of those metrics. In many cases, the massive changes affecting the defense industry have weakened the traditional means by which program and company histories are maintained. The movement away from strong functional organizations, for example, has eliminated many of the opportunities for cross-program and inter-generational knowledge transfer. Traditionally, it has been these informal means by which much development history and process knowledge has been maintained. While the use of the functional silos as pseudo apprenticeship programs may not be the most effective method of capturing technical knowledge, for many companies it is the only method in use.

Without a means of capturing the necessary historical information, the uses of metrics are severely limited. Just as a historical record of manufacturing process data is necessary to adequately assess production processes, the characterization of current product development processes relies upon a historical record of past development activities. It has been demonstrated in classic studies that the learning curve effect is more pronounced in highly human-intensive manufacturing activities than in automated activities.[15] In such a highly human-intensive process as design, similar learning can also be expected. However, due to the nature of design as a discontinuous process,

---

[15] Hirschmann, Winifred; *Harvard Business Review V42. No. 1*, "Profit from the Learning Curve"; Harvard University; ©1963

without an adequate design history to learn from, improvements on past design practices will be difficult to achieve.

In a similar fashion, the trend toward ever-longer development cycle times also may adversely impact the use of metrics in product development. Since the use of metrics relies heavily upon the availability of a historical record of past activities and much of that historical knowledge is stored in the memories of individuals within the development organizations, any lengthening of the time between successive developments would logically decrease the amount of knowledge available for future programs because of personnel attrition, corporate downsizing, or just plain forgetting of the original information. If the time lag between development programs becomes so great that this information is lost before the knowledge can be transferred to the younger generations of engineers, it is possible for entire program histories and the associated development knowledge to simply disappear from the company. Case B provides one example of just such a disappearance. That case involved the redesign of a portion of a system that had been deployed nearly two decades before. As might be expected, none of the original engineers were remaining with the company to provide insights into how or why many of the design decisions were made during the original system development. Although redesigns prior to the current one had also attempted to fix the problems with the system, they too had little original information to rely upon and had left little documentation of their own design processes for future designers to use. The current team was forced to start from nearly a clean sheet design.

Current Metrics Not Indicative of True Program Success

Although there has recently been a great emphasis on developing metrics to improve the product development process, in many cases they still are not accurate predictors of successful programs. Case A provides an example of a development program that was praised by its external customer for providing working test hardware while also meeting all cost, schedule, and performance targets; however, to the internal customer, the manufacturing division within the company, this design process was a failure because the product developed was not robust enough to be producible without a large amount of rework. Relying upon the development metrics alone to gauge the success of the program, in this case, was not sufficient.

The example of Case A demonstrates one of the problems with how the "success" of a product development activity is defined. There is a great temptation to define success as the satisfactory achievement of all of the programmatic goals: on schedule, on or under budget, and satisfaction of technical requirements. While on the surface this definition seems logical enough, it is flawed in one serious respect — it relies heavily upon the assumption that the program goals provide a complete and accurate indication of the actual customer objectives. While it is generally accepted that the true goal of any development program must be the satisfaction of the customer, there is no guarantee that simply meeting the contract requirements will be enough to satisfy all of the downstream customers actual requirements. In the portion of Case C being performed under a purely commercial contract, there is the understanding by the company and the development team that even if the eventual product satisfies the requirements detailed in the contract, if it fails to perform the function that the customer is expecting, the company will still be liable for "making it work".

This heavy reliance on the satisfaction of the strict contract requirements can potentially lead to a lack of adequate customer focus in the design process. This lack of customer presence extends down even to the level of the individual engineers on the design teams. In Case A, for example, when asked about the customer goals for the design program, one engineer responded with a list of the technical requirements for the component being designed. He remarked that these requirements were nearly the same as the previous generation of the same system and he did not understand why the customer would "buy something new if the old one can do the same thing"[16]. There was no customer interaction at that level of the organization to clarify of the top-level questions that the engineers had regarding customer preferences between conflicting requirements or to resolve trades between minimal performance gains and higher system costs. As one systems engineer put it, "(our engineers) could end up having to spend, literally, hundreds of thousands of dollars to meet some requirement that the customer, in reality, thought was a good idea, but if you went back to the customer and told them how much (it really costs) they'd say 'Forget it!'"[17]. Without being able to make the connection between the customer needs and the technical requirements of the design, the engineer has no way of adequately assessing the importance of those needs.

[16] Design Engineer, Company A; April 1994.

[17] Systems Engineer, Company A; April 1994.

5. ELEMENTS OF DESIGN DECISIONS

Good design metrics are one part of the overall development process. By themselves, however, even the best metrics can lead companies and design teams into making poor design decisions. The use of metrics in the decision making process is really linked to three closely-links factors: a sound decision-making process that integrates the disparate customer voices and requirements to achieve a product acceptable to all; the ability to measure and track the progress of the development team both within the individual programs and across different programs; and an accurate historical basis to upon which decisions can be based and to which the measurements can be compared.

5.1 Basic Elements of Decision Making Process

The interactive relationship of these three factors is presented in Figure 5.1. As shown, the information flows are bi-directional at all of the interfaces between the three factors. These flows can be viewed as the necessary inputs for each of the three elements. Additions to the historical knowledge of the company, for example, come from the record of past design decisions and the historical progression of the metrics tracked during the development programs. Likewise, at this high level, the primary inputs into design decisions come from the experience of the decision makers and the current state of the design program as indicated by design metrics. Each of the three elements and their relationship to the whole will be discussed below.

It must be noted that the information flow in Figure 5.1 is not intended to be a model of the entire design decision-making process, nor are the three processes represented implied to be the only processes important during the design. Other enabling practices such as the use of common terminology between functional groupings and cross functionally integrated databases are beneficial in achieving a good final design solution.[18]

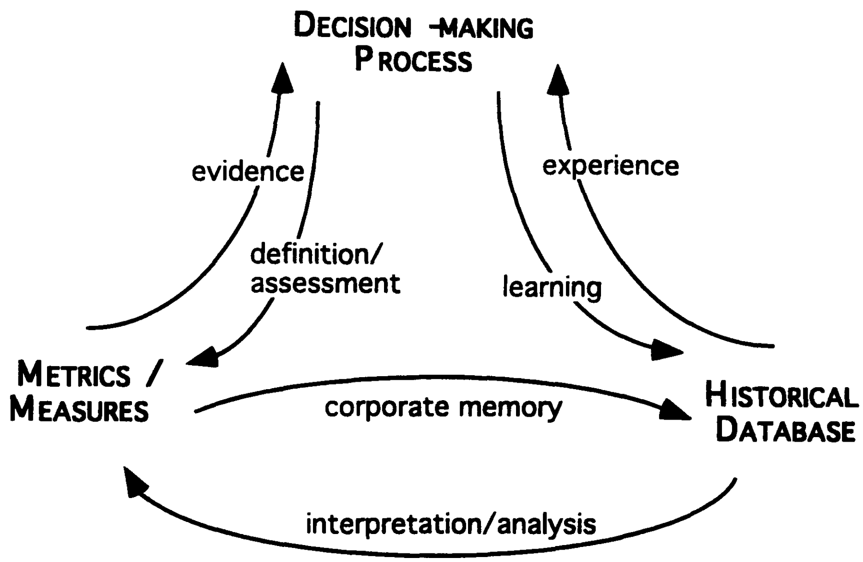

Figure 5.1: Three Elements of Product Development

5.2 Appropriate Metrics

The most evident of the three factors being considered are the design metrics. In most cases, metrics act as the principal feedback mechanisms during product development. The elements of cost, schedule, and technical performance factors, are used in some form by every product development group studied. Although the mechanism by which these metrics are tracked may change from company to company, the basic usage of the metrics is the same. Chapter 4 presented a framework for these metrics and the identification of where current metrics fall within that framework.

One of the hypotheses at the inception of this research effort was that the use of "good" metrics in the design process would provide significant improvement over the current practices. If companies were able to discover the proper metrics to use, they would be able to better understand and manage the product development process. While understanding the proper metrics to use is important, metrics alone will not solve many of the problems currently be encountered. Metrics are a small, but important part of the entire design system. By themselves, design metrics cannot address the difficulties of

[18] Hoult, David, et.al.; "Cost Awareness in Design: The Role of Database Commonalty"; MIT; ©1994.

product development; however, they are a vital factor in any good product development organization.

Metrics as Drivers of Design Decisions

As discussed in chapter three, the most critical of the primary metrics was the on-time delivery of the product to the external customer in all of the cases studied. This emphasis came primarily from needs the downstream external customers expressed through high penalties for schedule slippage written into the contract. Since a large portion of the final aircraft is produced by suppliers to the prime contractor, a culture of schedule dependence has arisen between the primes and their upstream suppliers. With detailed production and delivery schedules tightly linked across company boundaries, delivery schedule slippages upstream quickly propagate throughout the system. For the initial hardware deliveries there are zero inventories and often minimal delivery buffers. Combined with increasing cost pressures, these fixed delivery schedules can have the effect of forcing concessions in the quality or technical performance of the products being designed.

In Case A, for example, to lower costs the company reduced the number of engineering prototypes available for test and analysis during the development program. The fixed delivery schedule negotiated in the contract forced the engineers working on the project to make concessions in the amount of time that they had originally budgeted for testing and analysis of the hardware. This reduction of time pressured the engineering department into short cuts to meet the fixed delivery date. These short cuts came in the form of a less vigorous testing program of the preliminary hardware. Instead of testing the design hardware over its entire operational temperature range, for example, it was tested only at ambient and fixed extreme conditions. While the design was released to production on time, the more rigorous full-cycle testing program pursued during the production inspection revealed problems with the design over much of its required operating regime that had not been discovered during the design phase.

Although the original design plan had called for adequate numbers of test equipment for concurrent design of the hardware and manufacturing test equipment, when costs were unilaterally reduced to satisfy the customer demand for lower costs the amount of available test hardware became a primary constraint on the design. The selection of delivery schedule as the driving metric for the design resulted in decisions that both reduced the quality of the product and resulted in much higher overall costs to the program. Even though the development team succeeded in meeting both their individual cost and delivery goals, the product was not ready to enter manufacturing.

Overall, the focus on short-term costs and delivery schedule resulted in higher total costs and the necessity for an immediate redesign program to fix the problems with the original design.

Role of Metrics in "Lean" Product Development

A question arises as to what the proper role of metrics is in a lean product development environment. While it appears from the casework that the basic factors of cost, schedule and performance are not enough to ensure the success of development program, it is unclear as to what the appropriate measures should be. As described in chapter three, there is often a confusion regarding the types of "metrics" being used. The two basic types of metrics are benchmark metrics and tracking metrics.

The benchmark metrics, referred to by one company as "Lean Metrics", facilitate process improvements through the comparison of current practices to past practices and benchmarks of external companies or divisions. As such, these metrics are extremely important in the overall strategic development of the company and the people within it, but they are not normally suitable for use in the detailed decision-making within a specific design program. The benchmarks are predictors of how well the design team will do on a program, but say little as to how the actual design will be done. For that, the project-specific tracking metrics are more important.

Ideally, the tracking metrics used in product development flow directly from the customers' requirements for the product. In the current defense environment, however, those needs are often driven down to an extremely detailed level before the actual designers are ever brought onto the project. Engineers and designers rarely hear the actual voices or requirements of the people who will be using the product. Normally, these requirements are filtered through many organizational layers before they reach the design team. While some degree of filtering is necessary to restrict the sheer number of disparate voices driving the product design, the utter lack of customer focus at the detailed design level in some programs is astonishing. One of the engineers in Company A for instance when asked for the customer requirements of the hardware being designed was unable to give an answer. He knew what the design specifications that he was trying to achieve were, but was unable to translate those specs into any type of customer need. This gap between the customer and the designers was further explained by a systems engineer at Company A saying that the design teams just are "not interested in (interacting with the customers)". This lack of customer knowledge, however, feeds into another of the principal problems plaguing design decision-makers. With none of the original customer requirements to fall back on, the reasons behind many design decisions are lost soon after a choice is made and the design team moves on to another problem. Although there were generally detailed paper trails of the decisions themselves in the cases studied, the underlying thought processes behind the decisions were not documented well.

5.3 Historical Database for Product Development

While design metrics are important, they alone are not sufficient information upon which to base design decisions. A second factor essential input into the decision-making process is an accurate historical knowledge of past designs. This historical database[19] is useful both for assessing current design practices against those of the past and for helping to determine the risk of achieving specific product goals.

Definition of "Historical Database"

The historical database referred to here is not just a simple narration of product development activities in the past. While such a record might be interesting, it would probably not be especially useful to future development teams. To be useful in practice, a historical record must be accessible to the people who need the information it can provide. Throughout any program, volumes of data are generated about the program goals, design requirements and specifications, results of design trades, and general program history. While much of this information might be of value to future development teams faced with similar design problems, without a standard accessible means of storing the data, nearly all of it will be lost within a few years after the program ends. Traditionally, the historical data that this paper refers to has been stored inside the brains of the individual people within the functional organizations. In every company, there are functional gurus who carry with them a wealth of company design history and are consulted first whenever design problems crop up. With the current movement toward downsizing and reorganization away from functional silos, however, much of that hard-earned experience is being lost.

In none of the cases studied was a practical history or design knowledgebase available to the development teams. As the redesign group for a part originally fielded nearly two decades ago, the development team in Case B, in particular, was faced with a lack of adequate program history to draw upon. Having nearly no record of the decision-making process in the original design and only a limited knowledge of more

[19] Note that the term "database" in this paper is being used in it's most general sense and does not refer only to electronically-stored computer databases

recent redesign efforts, the design team began from what their original team leader described as a "truly clean sheet design". To overcome the initial handicap of limited knowledge about the product history, the team gathered program records and historical data from a wide variety of sources. To get insight into the failure history of the component, a product tester from the manufacturing labor force was added to the team. Additionally, members of past redesign attempts were included on the team to bolster the team's knowledge of what had been tried before, and a key supplier was brought aboard to provide insight into the manufacturing techniques of component parts. To get further data about the failure history in the field, the design team consulted flight line repair crews with first-hand experience of the hardware problems. Information from all of these sources combined to provide evidence of a failure mode that had not been considered important before. A detailed analysis of the product revealed that this fault was indeed a possible cause of the failures being observed, and the part was redesigned to eliminate the defect.

As this case illustrates, adequate historical records are usually available, but they are not in a readily accessible form, often being distributed throughout a number of locations both inside and outside the company. This inaccessibility severely limits their use for all but the most highly resourceful and motivated design teams.

Importance of History to Measure Improvement

Historical records are also extremely important in the use of benchmark metrics. Since these metrics are inherently comparative in nature, it is possible to define a "good" for them. In manufacturing for example, the defect rate, while a poor metric to characterize individual products, is a good metric for process benchmarking. Over time, the average defect rate can be measured for individual production processes and these rates can be plotted over a period of time. It is easy to see from such a plot whether or not the production process is improving (moving toward the optimal goal of zero defects) or worsening. Process benchmarking enables a manufacturing organization to achieve high product quality that otherwise would have to have been inspected into the product. The reactive inspection process has been replaced with the more proactive goal of process control and improvement.

From the cases studied, it does not appear that such a movement toward proactive design process improvement is occurring. While many of the companies visited had direct feedback on manufacturing process benchmarks to the individual workers, no such feedback was evident in the design centers. Design was, in some of the companies, very much an ad hoc process with little in common from one development program to the next. This lack of standardization in the design process makes benchmarking design a difficult task.

Use of History for Design

The historical record is also important when characterizing the size of the acceptable design space during the development process. Without an understanding of the current capabilities of the design and manufacturing organizations, it is impossible to get an accurate characterization of the risk factors affecting new product concepts. By evaluating the cost, schedule or technical requirements of the new design against the current process capabilities as defined by the historical performance record, a measurement of risk can be determined from the distance that the new design is outside the current capabilities space.

While experienced engineers may already have a feel for the just how far the new design is "pushing the envelope", these subjective feelings are generally difficult to quantify and hard to convince others of their validity. In Case A, for example, some of the mid-level engineers had expressed their concern about the reducing the funding level while trying to maintain the technical complexity and schedule of the initial program. They had felt that the cost-savings up front would be more than eaten up by cost overruns in later design or production. With no way to assess the degree of risk that the program was assuming by reducing the program up-front costs, the company went forward with the design even though the consequences of doing so ended up being a much higher overall program cost and more problems in the production phase of the program.

5.4 Decision-Making Process

Metrics historical data come together in the decision-making process within the development program. These decisions can actually be made at any level of the program from the individual engineer up to the corporate or even customer level. No matter the level at which the decisions are being made, the information flow is similar to that of Figure 5.1. In nearly all cases, the current status (evidence) is evaluated against a historical record or prior knowledge (experience), either explicit or implicit risk calculations are done and a final decision is made. If either of the two factors feeding into the decision-making process are inadequate, bad decisions will result. Likewise, if the actual decision-making process itself is poor, the best information in the world will not be enough to ensure good decisions. From the cases studied, three major factors were identified that adversely impacted the effectiveness of the decision-making process: misalignment between the individuals with decision-making authority and those with responsibility for the eventual decisions; requirements misunderstandings; and inadequate information feedback between the decision-makers and either the designers or customers.

Authority and Responsibility Incongruent

The first problem identified arises when the people with authority for making decisions impacting the design are not the same as those who have direct responsibility for the outcome of the design. Returning to the case of Company A, an example of mismatched decision-making authority and responsibility can be found in the decision to accept a reduction in the number of hardware development units. Rather than gaining consensus with the program engineers to determine how best to bring down the costs of the program, the number of and distribution of the development units was handed down to the project teams. To the engineers actually responsible for the development, this decision was interpreted as yet another management mandate that did not really consider the problems of the "real engineers". In this case, the decision makers avoided having to live with the consequences of their decision until near the very end of the development program when the design flaws were discovered during the initial manufacturing phase.

An example of a more institutionalized form of mismatched responsibility and authority is found in the matrixed organization of Company C. Within this company, the design teams are not independent of the functional organizations. Program budgets, rather than being controlled by the program managers, are allocated to the functional chiefs. This funding arrangement means that design team decisions are not funded until the affected functional directors agree with the decisions. Engineers, who are responsible both to the design team and their functional groups often have to revisit arguments previously settled in the design group when selling their design decisions to the functional bosses to get funding approval for the team. Program managers, the people most responsible for the success or failure of the development programs, do not have final budget authority over their own design teams. The primary benefit of this arrangement is that the people with the greatest functional knowledge in the company are able to review the details of the design at every phase of the program; however, this redundancy is also its primary weakness. The process of making nearly every decision twice can result in an incredibly slow pace to the design.

Requirements Definition and Flowdown Problems

In all of the cases, one of the principal problems identified by the engineers was that of requirements definition and flowdown to the design team. This problem can be traced to a general lack of communication between the individual links of the information chain from the customer to the design and manufacturing teams. One of the governing factors of this seems to be the number of links in that chain. As illustrated in Figure 5.2, the information circuit between the end-users and the designers is, at minimum, seven links. For suppliers to the prime contractor, this distance can easily grow to more than a dozen separate links that design information must flow through. This figure represents an idealized view of the information flow process. Within each link of the chain, there could be multiple organizational layers through which information is filtered and decisions made that affect the product requirements and overall design. The result of this entire process is many disparate and possibly unreliable voices of multiple customers must be condensed at the design team level into a final product that, most likely, will not represent what the original customer wanted very well.

Inadequate Information Feedback & Linkages

Another difficulty related to long chains of information flows is the problem of inadequate feedback on the design decisions being made. This lack of feedback is illustrated in Figure 5.2 by the unidirectionality of the arrows at nearly all steps along the information flow. Although great strides have been made within some companies to eliminate the design practice of "throwing it over the wall" to the next downstream customer, less has been done to eliminate this attitude between separate organizations.

Within both of the subcontractors of Case A and Case B there was cited a lack of strong, consistent customer support from the prime contractors. One engineer in Company A told of an incident early in the program where a customer's on-site representative had been involved in a discussion where Company A's engineers had been complaining about the poor quality of a specific piece of hardware that had been causing problems. A few days later, the customer came back to say that they did not want that specific unit because they had heard that "it was a dog". While there was nothing wrong with that specific unit that couldn't be fixed, the informal discussions between the engineers led to the rejection of a high-priced piece of hardware, and more importantly, server damage to the foundation of trust that had been developed between Company A's engineers and the customer's on-site representative.

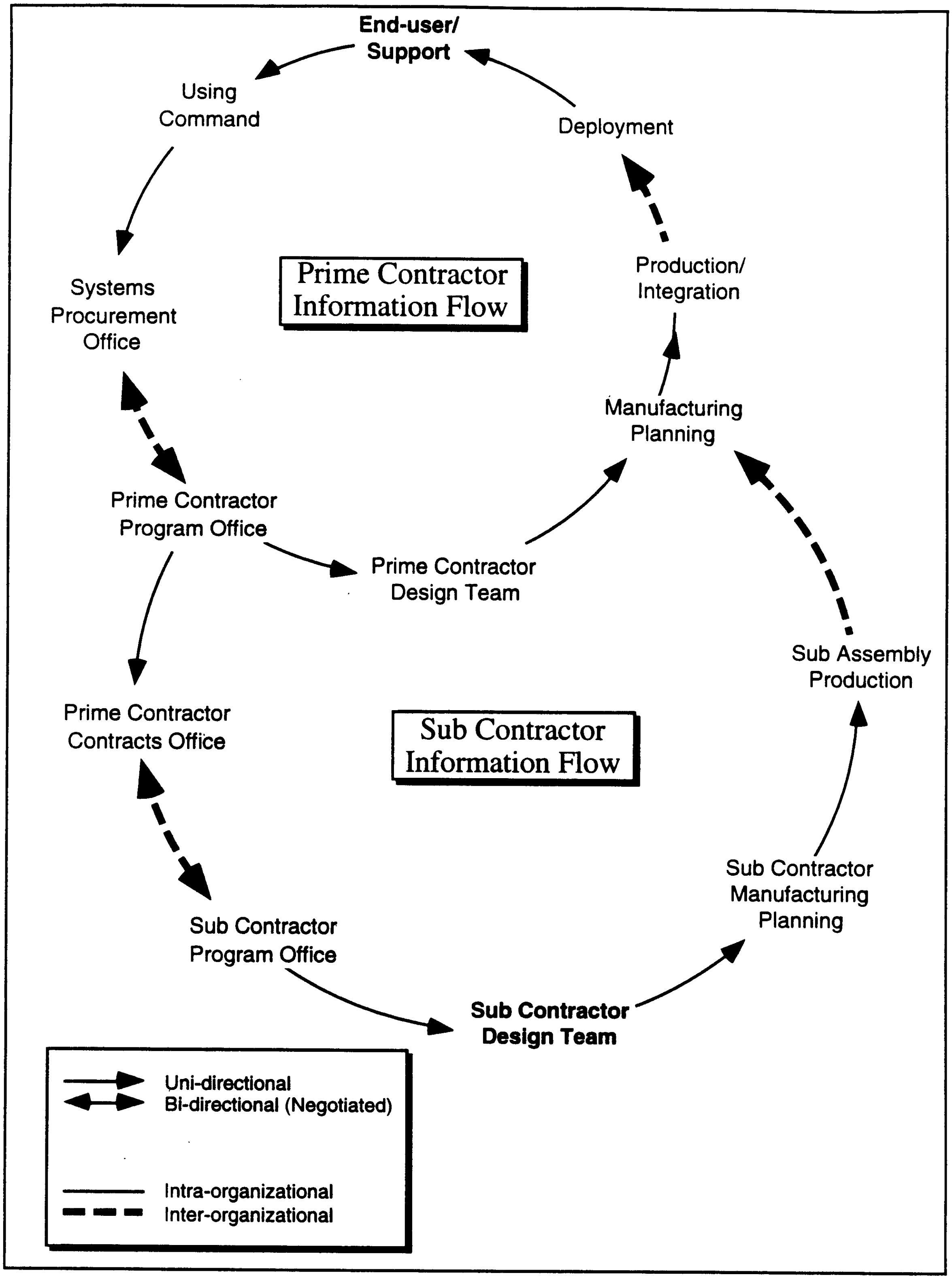

Figure 5.2: Design Information Flow

Case B reflects another aspect of the problem with inadequate feedback links between the designers and the customer. At the inception of the redesign program, the Company B got extremely good support from their primary customer. As the design progressed however, cost cut-backs by the customer reduced their participation in the design process by limiting the amount of time they could afford to spend for on-site at Company B. With less first-hand knowledge of Team B's progress, the customer's attitude toward the program shifted from one of cooperation and support to demands for immediate results. Their lack of participation in the design process allowed the customer to revert back into a more customary adversarial role and led them to suspect that Company B was not performing adequately.

Even within the companies, there are problems with the information flow between different parts of the design teams or functional organizations. One method of attacking these communications problems is the use of integrated product development for the program.

6. CONCLUSIONS

This chapter presents a short summary of the major conclusions of this thesis and suggests methods to address some of the shortcoming identified. Little emphasis will be placed on the examples and supporting evidence for these conclusions that has been presented in the previous three chapters. The three primary lessons learned from this research are:

- Metrics currently used in product development within the defense aerospace industry are not accurate indicators of the success or failure of development programs;
- Historical records of past development programs are normally incomplete or the information is inaccessible; and
- Design decisions often rely upon incomplete information and lack a strong understanding of the risks inherent in the design.

6.1 Metrics Not Accurate Predictors of Success

The original purpose of this research was to discover metrics that could act as up-front indicators of program success or failure by examining examples of current development programs from across the industry. Unfortunately, what was found was a lack of any such predictive metrics. The current metrics being used are generally reactive in nature and while useful for measuring some aspect of success, do not play a large role in driving the success of development programs.

Currently In Use: Cost, Schedule, Technical Performance

It was found that all of the metrics being used at the program level to monitor product development practices were combinations of the three primary metrics of Cost, Schedule, and Technical Performance. While other types of financial metrics were used to evaluate the company performance at the corporate level, these types of metrics had little formal linkage to the actual day-to-day activities within product development.

In the cases studied, the most common usage of development metrics was in retrospective tracking of the development program as it progressed. Cost, schedule, and technical performance factors were plotted against a development plan and variations were monitored to insure that corrective actions were being taken to "put the development back on track". The primary problems with such reactive measures were that they relied heavily upon the plan being correct and that they were unable to indicate potential difficulties until after they had already occurred — this can be compared to "driving a car by only looking in the rear-view mirror". While over-reliance upon a questionable schedule can be addressed through improving the scheduling process and a better understanding by managers that the schedule is only an idealized picture of reality, the problem of retrospective metrics is more difficult to deal with.

The case studies indicated that the more useful, but less common method of using development metrics was as proactive tools. These proactive uses could either be of the form of development process benchmarks or of using more reactive metrics during the program planning phase to better understand the upcoming development process.

Evidence from the cases suggests that the use of product development metrics as process benchmarks is not common. While metrics are used as planning tools by the development teams when preparing an initial schedule of activities for the program, actually comparing individual programs against each other or characterizing their behavior over time is inhibited by a number of factors. Most importantly, there does not appear to be such an obvious basis for comparison between different development activities as there is in manufacturing. Individual programs proceed on an ad hoc basis Although there is no evidence of causality from the case data, the one company that did have a more standardized approach to the development activity seemed also to have a better set of metrics for comparison between individual programs and for gauging development process improvements over time.

The most common proactive use of metrics is during the initial phase of the program where the development plans are originally being laid out. During this phase, insights are gained into the feasibility of the program plan. The metrics in this usage lend a numerical and visual reality to the suggested program plan. By determining what the metrics would do if everything went according to plan, the plan itself can be checked against the engineers and managers' intuitive knowledge of what the planned metrics should look like. This act of actually determining what the plots of the metrics should look like was felt to play a much greater role in facilitating the success of the program than did the actual tracking the metrics against the plan.

Metrics Not Drivers of Success

Although to some degree the success of the program is determined by the team's performance against their metric targets, there is little indication that tracking the metrics themselves actually drives the development team toward the goal of a successful product. While satisfaction of the metric goals might be one requirement for the success of the program, the metrics themselves are usually not enough to guarantee program success. It is entirely possible for a development program to satisfy all of its metric goals (cost, schedule and performance) yet still output a product that does not satisfy the actual customer need. This is primarily due to the fact that there is little feedback to insure that the metrics are actually derived from the customer needs. In the defense environment, many times the costs, schedule, and performance goals are determined well before the engineers on the design team become involved with the process. If conflicts between these requirements arise during the development process, designers often do not have sufficient knowledge of the customer needs to make an informed trade between them.

Reactive tracking metrics can report that a program is not succeeding, but they cannot provide insight into why a program is failing. A better indicator of the "whys" of program success can be found in the benchmark metrics. It is intuitive that good teams using good design processes are likely to result in good products. The difficulty is in identifying which teams and processes stand out as the best. Benchmarking of standard product development provides the necessary insight. By defining a standard product development practice that separates design into characteristic processes, metrics can be developed that are product independent and broadly applicable across many development programs. Without this type of feedback, the results of modifications to the design process cannot be quantified and refinements of the design process can be implemented to continuously improve the entire product development process.

Recommendations

One of the problems with relying heavily upon tracking metrics is that the customer feedback is not as direct in the defense environment as it is in a competitive market-driven industry such as automobiles. Whereas market analyses and sales figures for similar car models are primary inputs into the design of an automobile, in the defense aircraft industry much of this up-front customer input into the design has already been filtered through many layers before it penetrates to the level of the product designer. The functions that would be performed by a marketing organization within an auto company for example, are assumed within the customers' procurement processes.

This contributes to a lack of strong customer focus at the level of the design team. This lack of focus on the customer needs has contributed, in some cases, to an emphasis on the satisfaction of the metrics as an output of the design process rather than a means toward the greater goal of producing an actual product. The fact that any engineer working on a design would be unable to say what the customer wanted the product for is shocking.

Standard design activities should be identified and benchmarked to facilitate improvements in the process of design rather than just the output of design. Using an analogy from manufacturing to illustrate this point, great leaps in manufacturing efficiency result not from tight controls on the manufacturing output, but rather from continuous improvements in the manufacturing processes. You cannot reduce the number of defects by measuring the rate at which the defects are produced. To reduce defects, you must improve the quality of the process that is creating the defective parts. Similarly, tracking the output of the design process will not help to improve the process quality or efficiency. Focus on being good at design rather than just on creating good designs.

6.2 Historical Databases Inaccessible

The second major conclusion of this research was that historical information about past designs in many programs was surprisingly lacking. Even within ongoing programs, there was often little information about past decisions upon which to judge future design alternatives. When information was available, it was often stored in a form that was not readily accessible to the design team. The most common form of storage for this information was inside the heads of a few key people within the organization.

No Good Systems for Capturing Design Knowledge

Much of the knowledge of the organization is stored within the minds of the people within the functional areas of the companies in what one company referred to as "tribal knowledge"[20]. Unfortunately, in the current climate of downsizing, much of that corporate knowledgebase is in danger of being lost. Whereas traditionally knowledge has been transferred across program and generational boundaries within the functional organizations, in some of the new team-based organizations there is no formal mechanism by which functional histories are maintained. Even within the functional

[20] Term taken from discussions with representatives McDonnell Douglas Aerospace, March 6-8, 1995.

organizations, there is a danger of losing much of the tribal knowledge through normal employee attrition. On one development program investigated, a single individual was responsible for providing manufacturing knowledge to all of the design groups on a major aerospace project. His nearly thirty years experience in manufacturing and as a liaison to final assembly provides a unique knowledge base from which the designers can draw. At present, however, there are no plans to capture his unique insights for use on future programs after his retirement.[21]

A further complication to this movement away from functional knowledge bases is in how companies are approaching teams. Two of the three organizations profiled in the case work operated in a pseudo-product team environment with a relatively small core of "star" engineers moving from program to program within the company. At different points in the development program, different star players might be brought aboard to address the specific needs of the program at that time. This system creates a reinforcing feedback loop in which the stars gain a large amount of cross-program, but highly process-specific knowledge. This increases their attractiveness to other program at the same phase of development and the stars are shuttled from one program to the next as their process specialty arises. At the same time, however, the stars are rarely confronted with many of the downstream problems resulting from their upstream development work. This lack of downstream feedback has the unfortunate side effect of propagating systematic design problems throughout the company and leaving the non-"star" engineers with highly developed problem solving abilities but little cross-team functional knowledge.[22]

Additionally, with no formal method of capturing development histories, much of the knowledge gained during the course of a product development program is lost. While this might not be an important factor in some industries, the trend in defense aircraft is toward a larger portion of the business being upgrades of existing equipment rather than new aircraft production. With little or no history of the development process for the systems being upgraded, difficult and expensive lessons must be relearned by the upgrade development teams.

This lack of historical information within the design process leads companies to repeatedly run into many of the same problems from one development program to the next. Without a formal mechanism by which to compare the lessons learned from

---

[21] Personal interview with Dave Keeton, McDonnell Douglas Aerospace, March 8, 1995.

[22] Personal interviews with representatives of Companies A, B and C; April-June 1994.

individual design programs, it is difficult to get at the root causes of many of the problems being experienced in design.

Recommendations

There has been much written recently about capturing design knowledge in companies, but few good examples of methods actually in practice. One method of capturing design knowledge and facilitating organizational learning during product development put forth by Wheelright and Clark is the use of cross-functional project audit teams to assess the lessons learned by development programs[23]. This method identifies five categories in which knowledge can be gained from product development: Procedures; Tools & Methods; Process; Structure; and Principles. The project audit is a systematic retrospective review and analysis of a development program along to each of these five project dimensions by a cross-functional team. The audit is conducted very much like a separate design project whereby the audit team members have as their goal a review of the strengths and weaknesses of the recent development and recommendations for changes in future practices. This greatest limitation on this method is that it is retrospective in nature and relies upon an accurate program history to be successful. For extremely long development programs with high personnel turnover, such as those in the aerospace industry, there is a danger with this method of losing some of the more important lessons from early in the program.

Another type of organizational learning identified by Ward, et.al., was a method employed by Toyota[24]. Under this system, the functional areas develop short (10-12 page) "lessons learned" books that define the feasible design ranges for many different specifications. These ranges had been developed over many years of experience in determining producible designs. For the design engineers, the books act as a map of the feasible design space. Deviations from the specifications in the design books are noted and the affected functional groups must decide how best to resolve the problem. If the solution is to enhance the technological process capabilities of the organization, new acceptable design ranges are noted and the "lessons learned" books are revised. Through this controlled expansion of the acceptable design range, the number of feasible solutions to any given design problem is increased. One problem with this

23 Wheelright, Steven and Clark, Kim; *Revolutionizing Product Development, Quantum Leaps in Speed Efficiency and Quality*; The Free Press, New York, NY; ©1992.

24 Ward, Allen; Liker, Jeffrey; Cristiano, John; and Sobek, Durward III; "The Second Toyota Paradox: How Delaying Design Decisions Can Make Better Card Faster"; University of Michigan, forthcoming; ©1994.

method is that it might be difficult to apply in companies producing highly non-homogeneous products. This method might not be appropriate for companies where each new design represents a large expansion of the design space. Additionally, while this method captures technological process advances well, it does not address the structural or design process improvement that the project audit focused on.

6.3 Decision-Making Process Vital to Good Design

The third conclusion of this paper is that the actual decision-making process used within the development is a vital element in the overall success of the program. Although good program metrics and accurate historical data are important to the success of a design effort, without a sound process that combines both current metric and historical data when making design decision the project will probably fail. Two of the primary factors that were seen to drive bad decisions were: a lack of appropriate knowledge of the issues in question and a long organizational distance between the customers and the team making the design decisions.

Poor Metrics & Incomplete Knowledge Drive Bad Decisions

The primary driver of bad design decisions was an incomplete understanding of the decision to be made. This came about due mainly to either improper and poorly used metrics or a lack of direct knowledge of the project by the decision makers. The metrics most crucial for making design decisions are those that deal with the risks of different alternatives being considered. Currently risk measurements are a highly subjective art based primarily upon "engineering knowledge" which varies from one person to the next. Also, as has already been discussed in Section 6.1, the current design metrics provide little insight into the potential success or failure of any given design decision. For simple decisions where data is readily available the choice between competing alternative might be obvious, but for more strategic decisions such as how best to proceed with a design, choices must be made with little hard evidence to support any of the alternatives presented.

A natural reaction to this type of uncertainty is to allow decisions to be made by the individuals in the design teams who have the most direct knowledge about the situation. While this seems logical, the ingrained culture and hierarchical nature of many development organizations prevents critical decisions from being made at the design team level. Even when integrated product teams are used for the design process, on important decisions they are commonly forced to submit to greater authorities within the program or functional organizations. This practice led one engineer interviewed to characterize their company's organizational structure as "a team as long as the team decisions agree with management's decisions — as soon as we diverge, the team decisions get overridden." It is not unsurprising that this type of uninformed, unparticipative decision-making process results in bad design decision being made.

Large Distance Between Customers and Designers

One of the arguments against having important decision made at the design team level is that the engineers on the design team have little interaction with the customer and thus are unable to "see the big picture". Rather than being a justification of a poor decision-making process, however, this only illustrates another problem with many current systems. Because gap between the engineers and the customers is extremely large, there is no way for many of the low-level designers to accurately understand the needs of the customer. Since a great number of low-level decisions and design trades are made level of the individual engineer, it would seem important that the designers have a firm understanding of the needs of the customer when making decision that impact the product that will be delivered. This lack of customer focus at low levels in the decision-making process leads to a shifting of the design goals away from satisfaction of the customer needs and toward mere satisfaction of the contract requirements or program metrics.

The length of the information flow between the ultimate customer and the designers also leads to a problem of accurate flowdown of the program requirements. At every link along the organizational chain from the end-user to the design team, the information is retranslated into the specific language appropriate at each stage. The result of this extended game of "Telephone" is that the specifications reaching the design team rarely look anything like the original customer needs expressed in the field. This is most problematic for upgrade and redesign programs where the new designs must interface with existing hardware. Although the designers can be given a detailed specification of the interface constraints with which their designs must be compatible, without a strong understanding of the current systems and how they are used, making the new product truly integrate and work with the existing system is nearly impossible.

Recommendations

The most obvious recommendations here have been made before and there is little that can be done but repeat them again. Foremost, critical product decision should not only be allowed to be made at the design team level, but should be encouraged at this level. While such a practice goes against the traditional hierarchical culture within many corporations, creating a group design and decision-making process whereby design engineers are given both the responsibility for the success of their designs and the authority to make important decisions about those designs can have a positive effect on the overall design process. In the development of Chrysler's LH cars, for example, a more participative group decision-making process has led to better communication and a greater feeling of ownership of the design by team members from all design and engineering disciplines.[25]

Along those lines, customer needs must become the focus of all levels of the organization. Giving the design team authority over and responsibility for the design is one way of facilitating this, but this customer focus should also be made a more formal part of the design and development process. A more proactive use of basic quality function deployment techniques during the development program would help to more accurately deploy the voice of the customer throughout the organization and translate the customer needs onto the production floor.[26]

---

25 Scott, Gregory; *IMVP New Product Development Series: The Chrysler Corporation*; International Motor Vehicle Program Briefing Paper; ©1994.

26 Clausing, Don; *Total Quality Development: A Step-by-step Guide to World-Class Concurrent Engineering*; ASME Press, New York; ©1994.

7. NEXT STEPS

While this thesis has attempted to define and address many of the issues relating to the use of metrics during the product development process, there is much more still to be done. The purpose of this chapter is to identify specific issue that should be addressed either in further research or through initiatives within the companies themselves.

7.1 Topics for Further Research

This thesis intentionally took a very general approach toward the examination of product development metrics. The purpose of the research was as much to provide a foundation for further investigations as it was to address the specific problems faced by industry today. As such, a great number of questions were generated by the research for this thesis. Due to the practical constraints on time and resources, however, many of the questions raised are unanswered within this document. This section identifies some of the additional research that is necessary to complete the job begun by this thesis.

Validation of Thesis Conclusions

The first undertaking should be a further validation of the major conclusions of this paper. Since much of the evidence for these conclusions came from a limited number of case studies, the general applicability of these conclusions is suspect. While the research was designed to allow for the inherent limitations of case studies, until a broader, more detailed survey of the industry can be performed, the general applicability of these conclusions cannot be guaranteed.

Further research is necessary to identify best practices in such topics as the maintenance and usage of historical data and design team empowerment, and to examine methods by which such practices could be disseminated throughout the industry. Ideally, such identifications should come from detailed, on-site surveys of each of the companies involved to assess the current practices and determine potential best practices. While this ideal situation is probably impossible to attain, it might be reasonable to attack this research through a focused questionnaire designed to pinpoint best practice candidates followed up by more complete on-site analyses of the candidate companies.

Maintenance of Knowledge Base

Many questions are raised regarding the issue of corporate learning and the maintenance of the existing knowledge base. Much has been invested over the past four decades to build up a military production industry second to none in the world. Now with the end of the cold war and the shrinkage of the military-industrial complex, there is a serious question of whether to preserve the current military capabilities. If it is decided that such a preservation is in our national interest, how best can we maintain these capabilities in the face of shrinking procurement budgets.

Whereas traditionally much of the development knowledge within the companies has been stored within the human resources within the organization, today such a policy is not viable in the long run. Corporate consolidation and downsizing combined with the increasing length of procurement cycles has resulted in a situation where the number of people within the organization who are present throughout the entire lifecycle of a product is dwindling. Without such people to pass what they have learned on to future generations of design engineers, the knowledge of the company is in danger of being lost.

A detailed study on this topic would address the questions of how can current organizations facilitate learning; what type of historical data is important to capture; and what is the best way to store and manage such information. Additionally, the applicability of such practices as the use of Toyota-style lessons learned books to the aerospace industry should be considered. These studies should take into consideration how learning can be facilitated given both the highly technical nature of this industry as well as the current trend toward lower procurement levels and reduced development activity.

Length of communications/decision making hierarchy

This thesis concluded that the length of the decision making hierarchy played a major role in how effectively design decisions were made. A further characterization of this role is in order. While great strides have been made within some companies to realign into flatter organizational structure and reduce redundant levels of management oversight, there still appears much to be done. From the examples presented in this paper, it is apparent that in many cases the designers and assemblers are still too far removed from the customer to maintain the link between the customers' needs and the product specifications. Traditional rigid hierarchical structures that lengthen the communications chains and hamper cross-organizational communication still exist within many companies and the military services.

Further study into this topic should address how better to use existing tools such as Quality Function Deployment (QFD) and the House of Quality to facilitate greater levels of communication throughout the design process. While these tools do provide a great potential benefit, there is little indication that they are being employed to their fullest extent within this industry. A study of how they have been implemented at other, non-defense, companies might be helpful in understanding what barriers exist to their implementation within the defense industry.

7.2 Development Issues To Be Addressed

Throughout the course of this research, issues were raised that were outside the limited scope of this thesis. Many of these issues, however, are worthy of additional analysis and are presented in this section as general suggestions for further research or bases for pilot programs.

Measuring "Success" in a Defense Environment

One of the primary questions raised by this research was that of just how to define the success of a product development program. The case studies in this thesis indicated that development success was often defined in terms of easily measurable quantities or factors. If a program satisfied the contract requirements, it was usually considered a success even if the results were not acceptable to the immediate downstream customers or failed to address the true customer needs. For defense procurements, the burden of addressing those needs fell not onto the development team, but was wholly contained within the procuring organization.

There are indications from anecdotal evidence that incentives within the industry favor current practices over more Lean methods. Much of the cost and profit motives within commercial companies simply do not exist within the current defense environment. Implicit incentives favor long development cycles, underbidding, and poor quality in both manufacturing and design. Even when programs fail to satisfy contract requirements, often the deficient companies are often awarded additional funds to redesign the deficient parts to bring them into compliance with the original specifications.

An analysis of the incentive structure for product development programs could provide insight into how development programs can be a "success" without truly succeeding.

Role of Manufacturing Concurrency & Prototyping

An additional ancillary issue that has arisen from both this research and other studies within the initiative has concerned the role that prototyping plays in the development process within defense companies. While in Lean organizations, prototypes have been used as process debugging tools, from the evidence seen in the defense industry they primarily represent proof of concept vehicles for the engineering departments. Even in cases where manufacturing personnel are consulted or actually employed in the development and production of the prototype hardware, often these prototypes are manufactured using methods different from those that will actually be used during the production phase. While insight gathered by bringing manufacturing gurus into the development program is important, this type of relationship is not enough to bring out some of the more complex problems to be encountered during production. Even the most experienced manufacturing engineer or assembly worker often cannot predict many defects until the units actually reach the production line. While most companies would find it appalling to send equipment into manufacturing before the functionality had been tested, they see nothing wrong with implementing untried production processes at such a late stage. Without both a technically acceptable product and a working production process, the design cannot be considered complete because no products can be delivered to the customer.

Management of Government Furnished Equipment

One issue that came directly out of the research for this thesis was that of the use and management of government furnished equipment (GFE) within the defense companies. GFE is parts, supplies, and capital equipment supplied by the government to be used in the production programs on government contracts. This material is lent to the companies during the production phase, but remains the property of the government throughout the program. While this arrangement reduces the risk and the costs incurred by the company on any given program, there are indications that this arrangement is not being exploited to its maximum efficiency. For commercial companies, capital equipment can be moved between different programs as necessary; however, defense firms are restricted to using this program on the specific programs for which it was purchased unless they obtain approval from the SPO to move it to another.

Unfortunately, this means that equipment owned by the government often languishes underused in one program while another program, sometimes on the same assembly floor, is forced to procure nearly identical equipment (at the government's expense) simply because they cannot have access to the existing tools.

There was the further implication that the primary obstacle to a more efficient distribution of GFE within and between companies was primarily the lack of any central database of GFE purchases and usage within the government. One engineer interviewed during the research for this paper made the comment that their company "knew where the government's stuff was better than they did!" After all, the companies are required to keep extensive records on the location and usage of any GFE they maintain. If these individual records could be managed in such a way as to allow the movement of flexible GFE machinery between programs (or between companies) simply, the potential exists for significant savings throughout the defense industry.

7.3 Final Comments

This thesis represents only a very small portion of the entire work that still must be done to make the companies of this industry competitive into the next century. As important as these first steps are, it is the follow-on work that will truly determine whether and how companies will survive in the coming decade. Already this industry has seen a massive restructuring and consolidation as government military funding has been reduced. Even within the lifetime of the Lean Aircraft Initiative, a number of companies have come and gone as business units have been merged, bought or sold by others within the industry. While some pain is inevitably in transitions of the type that this industry is experiencing, it can be hoped that the organizations that do survive will be stronger and better-prepared to face the lean years ahead.

The future belongs to those who are capable of responding swiftly to meet changing customer needs and who can evolve rapidly to the unstable defense environment that will characterize the coming years. While it is possible that simply being Lean will not be enough to guarantee survival in this new age, those not Lean will almost assuredly perish. The race has begun, and only time will separate the victors from the vanquished.