A Systems Engineering Approach to Improving Vehicle NVH Attribute Management

By

Michelle Lorraine Sacka

B.S. Mechanical Engineering
University of Michigan, Ann Arbor

Submitted to the Department of System Design and Management
In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

at the

Massachusetts Institute of Technology

January, 2008
[February 2008]

Signature of Author..
Michelle Lorraine Sacka
Department of System Design and Management
January, 2008

Certified by..
Daniel E. Whitney, PhD
Senior Research Scientist – Center for Technology, Policy, & Industrial Development
Thesis Supervisor

Accepted by..
Pat Hale
Department of System Design and Management Director

This page is intentionally left blank.

A Systems Engineering Approach to Improving Vehicle NVH Attribute Management

By

Michelle Lorraine Sacka

B.S. Mechanical Engineering
University of Michigan, Ann Arbor

Submitted to the Department of System Design and Management
In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

ABSTRACT

This research is comprised of a detailed study of attribute management processes at a North American Automotive OEM (NA OEM) that has just introduced a new product development system intended to drastically reduce product cycle time and expedite product time to market. In specifics, the product development processes and organization that manage the delivery of a vehicle system design that meets or exceeds customer expectations for noise, vibration and harshness (NVH) are studied. Systems engineering principles, methods and tools are applied to the current processes to assess if process lead time, resources and product quality improvement can be realized. The systems engineering Design Structure Matrix (DSM) method for product development process modeling is applied to the current process used to manage the highly cross functional vehicle attribute known as second order NVH. Second Order NVH represents a vehicle system attribute that is owned by a single subsystem, yet controlled by design parameters owned by many other subsystems. The DSM method enables the NA OEM PD organization to understand the current process of managing this highly cross functional attribute and serves as a powerful tool for process restructuring. Process data is collected such that the DSM process model can be input into a simulation program which predicts stochastic process lead time for the current process and tests the impact of process restructuring ideas. This research also studies the methods and tools used at NA OEM to facilitate vehicle attribute trade-off, decomposition and cascade to the subsystem and component level. Then, a systems engineering approach is suggested to improve the attribute engineering knowledge base which could enable improved attribute trade-off, decomposition and cascade.

Thesis Supervisor: Daniel E Whitney, PhD
Title: Senior Research Scientist, Center for Technology, Policy, & Industrial Development

ACKNOWLEDGEMENTS

First, I would like to thank my thesis supervisor, Dr. Daniel E. Whitney, Senior Research Scientist at the Massachusetts Institute of Technology. Dr. Whitney's enthusiasm for and depth of experience in the field of systems engineering enabled me to gain a much deeper understanding and appreciation for this developing field of engineering. His many years of experience working with automotive OEMs brought many invaluable insights to my work. He continually challenged me to rethink the way things are currently executed at various automotive OEMs. In addition, his prompt and thorough feedback enabled me to bring this work to completion.

I also appreciate valuable guidance from the following people who have made previous contributions to the field of systems engineering; Antonino Zambito, Qi Hommes and Jehanzeb Noor. They all took time away from their busy schedules to speak with me in person about their previous work.

I would like to thank the following managers from the corporate alliance for their support and guidance despite their challenging workloads; Dana Katinas, Terese Bertcher, and Amir Keyvanmanesh. I would also like to thank all of the product development engineers who made valuable contributions to the data collected for this thesis.

Most of all, I would like to thank my personal support system because without them this work could never have been created. I would like to thank my Mother-in-law and Father-in-law Dianne and Timothy Sacka and my mother and father Nancy and Robert Rogers. They truly held me together as I took on the challenge of creating this thesis amongst many others. Most importantly, they spent many hours giving our newborn son, Jacob, the loving care he needed as both my husband and I pursued our master's degrees after hours. I would also like to thank my husband, Tim Sacka, for his continuous support, encouragement and positive outlook.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Motivation

Over the past three decades the automotive competitive landscape in the U.S. has changed drastically as market share has shifted from U.S. automakers to foreign automakers. In the first quarter of 2007, it was reported that the "big three," referring to GM, Ford, and the Chrysler Group, barely held over 50% of the U.S. market share combined. In the first quarter of 2007, each of the big three NA OEMs saw an overall decline in both U.S. sales and market share compared to one year prior. In contrast, the large import OEMs, Toyota, Honda, Nissan, all saw an increase in U.S. sales and market share in 2007. [Teahen, 2007]

Global competition is forcing Automotive OEMs to rethink competitive strategies for all aspects of the value chain. Automotive OEMs are resizing the company to match diminishing market share, restructuring their operations, reshaping their supply chain, and striving for cost competitive component pricing. However, the one area within these enterprises that has tremendous potential to provide competitive advantage is their product development organization and processes. [Smith & Eppinger, 1997] After all, the Product Development organization within OEMs is tasked with creating a product that meets or exceeds customer expectation, is delivered to the market on time and within budget and maximizes value to all stakeholders in the automotive value chain.

Currently, automotive OEMs are focused on reducing their product development cycle time as well as drastically cutting their work forces. In 2006, a top NA OEM executive declared:

> ...We're also speeding our product development time and improving time to market between 30 and 50 percent by the end of 2008. And, in 2009 and beyond, the product onslaught accelerates even further... We've reexamined our entire cycle plan, and we've accelerated work on future products. [Larkin, 2006]

In that same announcement, the OEM executive announced:

...In line with this new reality, we will resize our business in North America. That includes reducing our total annual operating costs by about $5 billion by the end of 2008. As part of these cuts, we will reduce our salary-related costs by about a third, or about 14000 equivalent salaried positions. [Larkin, 2006]

Automotive companies are no longer afforded leisurely product development cycle times where products are conceived and brought to market in four or five years, often over budget. Nor can automotive companies afford to allow quality issues to reach the customer. Today there is tremendous "do or die" pressure to create higher quality products in less time with fewer resources. It seems as if every aspect of this task defies the "iron triangle" where project resources, timing and scope fight each other and the success of a project cannot be guaranteed if these project dimensions are not carefully aligned and balanced. [DeWeck, 2006] Thus, in order to achieve success, automotive OEMs must look inwardly at their product development process organization and ensure that all effort and time expended on product design add value to all stakeholders in the value chain. This means there is no room for waste in the product development process and the output of the product development processes must always meet or exceed customer expectations.

One North American OEM, referred to as NA OEM throughout this thesis, has recently adopted a new product development system, referred to as New PDS, aimed to significantly reduce product cycle time from concept to launch. The reduction in product cycle time is achieved by a reduction in the total number of prototype builds and an increase in concurrent engineering. However, as the new leaner product development process is rolled out, some engineering organizations within product development are finding difficulties meeting milestone deliverables and timing.

One contributor to delayed product development milestones, budget run-overs, and quality issues is the inability of some vehicle system and subsystem engineering organizations to produce a design "on time" and within budget that meets vehicle level attribute objectives. Attributes are defined as vehicle level quality inherent in the subsystem design and integration. Customers experience attributes as either positive or negative. Negative attributes are commonly referred to as error states. Attribute are further defined in section 3.3. In some cases, issues with vehicle attributes not meeting the customer defined objectives are allowed to occur as late as vehicle launch. So, the question exists, how can an automotive OEM product development organization, tasked with reducing both their product development cycle time and their resources, deliver vehicle designs on time with attributes that meet or exceed customer expectations? The use of systems engineering principles, methods and tools within the product development organization can play a significant role in enabling the vehicle system to meet attribute objectives as well as in reducing total time and resources to deliver these designs.

1.2 Problem Statement

The automotive OEM's product development processes must yield a vehicle design that meets or exceeds customer expectations. In addition, the vehicle design must be delivered to the customer "on time" and the cost to produce that vehicle must be minimized. Ulrich and Eppinger define five dimensions that characterize a successful product development effort. These five dimensions are product quality, product cost, development time, development cost, and development capability. If product development efforts are failing in any of the five dimensions defined by Ulrich and Eppinger, then the existing processes should be carefully examined and corrective actions taken. [Ulrich & Eppinger, 2004]

In today's competitive environment, automotive OEMs are finding it increasingly difficult to deliver vehicles with inherent attributes that meet the customer's high demands. How does the vehicle feel and sound when driven in all conditions? How many miles per gallon does the vehicle get? Vehicle attributes such as ride, handling, performance, safety, sound quality, and fuel economy can provide a competitive advantage for automotive companies. The product development processes must enable successful integration of all vehicle subsystems to create a vehicle with attributes that meet customer expectations. It is important to note that competitors continually strive for vehicle designs that achieve attribute performance levels that raise customer expectations. Therefore, an OEM's product development system must also continually incorporate changing attribute targets.

A systems engineering approach, holistically focused on improving the product development processes and the organization, may help enable vehicle systems level attributes to meet customer expectations on time and within budget. A systems engineering approach is needed because the vehicle system can be considered complex with thousands of components that are integrated into many subsystems which are then finally integrated into the final vehicle system. These systems and subsystems are responsible for delivering the attributes. In some cases, attributes are delivered by combinations of subsystems. Moreover, the integration of vehicle subsystems is not purely hierarchical as vehicles subsystems are not purely modular. Rather, vehicle subsystems have many interfaces with energy, material and information flow. System level attributes are extremely coupled. In addition, mass production of vehicles adds another dimension of complexity. These complex systems must be replicated at high volume and maintain predictable functionality and performance within manufacturing capabilities.

As a starting point, OEMs can focus on improving the delivery of vehicle designs that affect "high impact" vehicle attributes by using a systems engineering approach to PD process and organizational improvements. This thesis defines "high impact" vehicle attributes as those that have historically caused issues at vehicle launch and are one of the top ten warranty issues for high volume vehicles and attributes. High impact vehicle attributes, such as noise, vibration and harshness (NVH), emerge as a result of how cross-functional engineering organizations design and integrate their subsystems into the total vehicle system. Attributes, such as NVH, can only meet customer expectations if interfacing vehicle subsystems are designed and integrated correctly and "on time." OEMs can first identify "high impact" attributes. Then, OEMs can study the current processes used to trade-off, decompose and cascade attribute targets to the systems and subsystems that deliver them and then deliver a design that meets the targets for those vehicle attribute. OEMs can use system engineering principles, methods and tools to examine these attribute management and product development processes and deploy changes where required. Systems engineering principles can be used when trading-off and decomposing system level attribute targets into subsystem and component level targets. Also, system engineering process modeling methods, such as the design structure matrix and associated simulation tools, allow an organization to understand their current product development processes at any chosen level of detail and propose and test improvements to this process.

However, when focusing on a single or a few chosen attributes, the systems engineer must be aware of attribute coupling and trade-off based on attribute prioritization.

1.3 Thesis Scope

This research tests the proposal described above by focusing on a single North American Automotive Company, name not to be disclosed and to be referred as NA OEM. This research focuses on the current product development processes and organizations that contribute to a vehicle system design that meets the attribute target for the attribute referred to as second order noise, vibration and harshness (2nd order NVH). 2nd order NVH has historically been problematic at vehicle prototype builds and product launch. Moreover, 2nd order NVH currently causes a high incidence of warranty claims. Data is collected on the current product, processes and organization through a series of interviews with engineers and managers from multiple organizations that affect 2nd order NVH vehicle response. Also, interviews are held with internal experts who adapted and deployed the new product development system recently introduced at NA OEM. Current process and organizational documentation is also reviewed. Specific process information is collected to create a design structure matrix (DSM) that models the current process flow for subsystem design, integration and delivery of a final vehicle design that meets 2nd order NVH attribute targets. This process information includes a list of tasks, the associated deliverables and responsible engineering title, stochastic task times, probability of rework, task sensitivity rating and information variability rating. This information is used to create a DSM simulation model of the process and establish a baseline for total development time. Then, the DSM and the simulation are used in conjunction with additional interviews with engineers to establish process improvement ideas derived from a systems engineering approach. The DSM process simulation model is used to determine the approximate percent improvement that might be realized by the implementation of these ideas. Additionally, the concepts of attribute target trade-off, ownership and cascading is discussed and systems engineering methods are suggested to ensure that attributes are prioritized and traded-off against other attributes, owned by appropriate systems engineers armed with appropriate systems engineering tools and cascaded to and negotiated with subsystems.

The purpose of this thesis is to serve as a template and framework for product development organizations that desire to study an area of their product development process in detail in order to improve product attributes and minimize product error states (negative attributes). The systems engineering approach, presented in this thesis, is not limited to the automotive case study presented in this thesis.

1.4 Thesis Content

Chapter 1: The first chapter of this thesis introduces the motivating facts behind the creation of this thesis and presents a detailed problem statement, scope and content description. The current state of the North American automotive industry is discussed. An argument is presented for why improved product development processes can provide a competitive advantage and how systems engineering principles, methods and tools can be used to improve PD processes which will ultimately lead to improved vehicle system level attributes.

Chapter 2: Presents an overview of literature that was reviewed for the creation of this thesis work. Literature was reviewed from the fields of Systems Engineering, Systems Architecting, Product Development, Product Development Process Improvement and Restructuring, The Design Structure Matrix, Attribute Management, Attribute Decomposition and Knowledge Management.

Chapter 3: This chapter provides an introduction to systems engineering and to system level Attributes. Common issues found in systems engineering with attribute trade-off, decomposition and cascade are discussed.

Chapter 4: An introduction to the thesis case study is presented. This chapter describes the vehicle level attribute referred to as second order noise, vibration and harshness. Also, all subsystems and corresponding subsystem engineering organizations that contribute to this vehicle level attribute are introduced and discussed. This chapter discusses why the processes and organizations that control this attribute must be modeled, studied, and improved.

Chapter 5: This chapter introduces the Design Structure Matrix (DSM) process modeling systems engineering method. The benefits that an activity based DSM can provide an engineering organization are discussed. Then the method used to create a DSM model of the current state of the product development processes used to deliver vehicle designs that meet specific vehicle level attribute targets is discussed. Major DSM analysis concepts are discussed and DSM terminology is defined here.

Chapter 6: This chapter presents the application of the design structure matrix system engineering method to the thesis case study based on how engineering organizations within NA OEM PD engineer and integrate subsystems that contribute to the high impact vehicle level attribute known as second order noise, vibration and harshness. First, the rationale for creating a DSM that models the current second order NVH attribute management processes is discussed. A history of issues found late in the design process as well as a high incidence of warranty claims helps to classify this attribute as high impact. The DSM creation process is discussed and the "As-Is" DSM is presented. Insights from the DSM creation process are presented. Then, DSM analysis techniques are used to create process improvement recommendations. These process improvement recommendations are consolidated into a new "to-be" DSM. Current excel macros written to assess total process time using the DSM are used to evaluate the approximate percent improvement in product development time. Then a qualitative discussion is presented for how the new process will improve product quality.

Chapter 7: In this section the current attribute target trade-off, ownership and control, decomposition and cascading at NA OEM are discussed and a systems engineering approach to attribute management improvement is proposed.

Chapter 8: Here general conclusions are presented about how systems engineering principles, methods and tools can help to improve the product development processes and organization and ultimately yield a higher quality product in less time. Process and organizational recommendations are presented. Future areas of research are discussed.

2. LITERATURE REVIEW

2.1 Product Development Organization and Processes

In their book, *Product Design and Development*, Karl T. Ulrich and Steven D. Eppinger explore the fundamentals of product development processes. They discuss the characteristics of a successful product development process. They maintain a holistic view of the product development process by discussing the participation of all core functions within a firm. The challenges of product development are explored. They stress the importance of cross disciplinary communication and decision making. Finally they present a useful framework for firms of all sizes that are developing any type of product. The framework consists of a structured method for completing product development activities as well as useful templates for major product development decision making activities. [Ulrich & Eppinger, 2004]

James Morgan and Jeffrey Liker present a detailed account of the inner workings of Toyota's Product Development Organization and Processes in *The Toyota Product Development System.* They discuss the key aspects of Toyota's Product Development system that have enabled Toyota to expedite its product development cycle time beyond its competitors as well as maintain level of product quality above most competitors. Morgan and Liker focus on value stream mapping as a vital tool for the product development improvement process. Toyota applies lean principles to product development where the product that is transferred throughout the process is information. Toyota focuses at the highest level, on improving the entire product development value chain as opposed to low level local optimization. There is tremendous focus on the application of common system architecture and principles of reuse. Moreover, Toyota stresses the necessity for design activity discipline to get the design work done early in the process. Toyota stresses that upper management need to create an organization and environment that enables a successful product development process. [Morgan & Liker, 2006]

Steven D. Eppinger, Krishnan Viswanathan, Daniel E. Whitney, Robert P. Smith, and Tyson R. Browning have completed extensive work in the area of Product Development. Their work stresses the impact that a firm's product development organization and processes have on the firm's overall competitiveness. In their many published papers, they explore principles, methods and tools for improving a firm's product development organization and processes. Robert P. Smith publishes a study that better defines iteration in product development. [Smith, 1998] Smith and Morrow publish a critique of product development process modeling. [Smith & Morrow, 1999]

2.2 Systems Engineering, Design Structure Matrix and Attribute Management

Systems engineering is a developing field of engineering. However, many insightful resources on the topic of Systems Engineering exist. The international council on systems engineering (INCOSE) offers the latest information and developments in the field of systems engineering. INCOSE has developed and maintains the "Guide to the Systems Engineering Body of Knowledge" also known as the "G2SEBoK." This Guide is available online as a resource for understanding the practice of Systems Engineering. The INCOSE website is also home to the Systems Engineering Tool Database. This database contains a searchable catalog of all tools that may assist in systems engineering problems. [INCOSE, 2007]

In addition to INCOSE resources, the US Department of Defense (DoD) has created a guide to systems engineering, "Systems Engineering Fundamentals." This work provides a conceptual level description of systems engineering and life cycle management. It also provides a framework for planning and assessing complex systems development. [DoD, 2001]

In "The Art of Systems Architecting," Mark W. Maier and Eberthardt Rechtin offer principles methods and tools for architecting complex systems. They make a distinction between architecting and engineering by stating that architecting deals largely with unmeasurables using non-quantitative tools and guidelines based on practical lessons learned. Whereas, engineering deals with quantitative tools and technical optimization. However, systems architecting serves as the foundation for systems engineering.

Donald V. Steward adapted the n^2 matrix to develop a Systems Engineering tool known as the Design Structure Matrix or Dependency Structure Matrix (DSM). Since then, the DSM has been adapted to many critical systems engineering uses and has been used to study and improve engineering systems and processes. [Steward, 1991] Smith and Eppinger offer a model that estimates iterative process lead time. [Smith & Eppinger, 1997] Krisnan, Eppinger, and Whitney expand upon the concept of concurrent engineering and present a study on how and when tasks within product development can be overlapped [Krisnan et al. 1997]. Yassine, Zambito, Lavine, and Whitney present case studies that demonstrate that a depth of knowledge in the process being studied with the DSM can provide more process restructuring benefit than pure algorithms alone [Yassine et al., 2000]. In addition to these technical papers, the official DSM web site, DSMWEB.ORG, offers valuable links to DSM references and tools [DSMWEB, 2007].

Several theses, published at MIT, offer new applications of the DSM and expand upon its capabilities. Antonino Zambito [2000] uses the concept of estimating probability of rework by determining dependency sensitivity and task variability to compute an overall dependency volatility. Soo-Haeng Cho [2001] compiles the work of Eppinger and Smith to create an Excel based program for analyzing activity based DSMs and computing process lead time and standard deviation. Eric McGill [2005] creates a Matlab based program to restructure DSM process models and to compute process lead time and standard deviation estimates based on stochastic task time, rework probabilities, rework impact, learning curve and resource allocation. Jehanzeb Noor [2007], focuses on the use of DSM process modeling and simulation to better manage attributes of complex systems.

Noor's research on systems engineering's role within a North American Automotive company brought many insights to the research and work in this thesis. Noor presented a systems engineering approach to improving the process of attribute management within the closure[1] system organization. This systems engineering approach consisted of both process and organizational improvements. His work explored the complex coupling between many of the closure system attributes. He presented several systems engineering methods and tools for managing these attributes. These methods and tools included the design structure matrix, datum flow chain, and axiomatic design. The research in this thesis will expand upon Noor's work by exploring a single highly cross-functional vehicle attribute that is highly coupled with other cross functional attributes. [Noor, 2007]

[1] Closures are doors, hoods, and trunk lids.

3. SYSTEMS ENGINEERING & ATTRIBUTES MANAGEMENT

3.1 A Systems Engineering Approach

Today's automotive vehicle system, product development processes and organization can be classified as a complex engineering system, as it meets the classification criteria for a complex engineering system set by academics developing the field of systems engineering. The criteria to classify an engineering system as sufficiently complex as to require a systems engineering approach are the following [DeWeck, 2006]:

- Numerous components and interconnections, interactions or interdependencies that are difficult to describe, understand, predict, manage, design, and/or change.
- Designed by humans having some purpose; large scale and complex engineering systems will have a management or social dimension as well as a technical one.

Thus, it is critical for the automotive product development organization to employ Systems Engineers who are armed with both a depth of product knowledge and a formal training in systems engineering. The Systems Engineer must work within a systems engineering framework and use chosen systems engineering principles, methods and tools as they guide the product from concept development, through system design, and finally to system life cycle management. And these system engineers must have management support, especially if they are working across organizations.

Figure 3.1 represents the primary components of an effective systems engineer. The head is represented by systems engineering principles. A Systems Engineer must become a systems thinker, know how to classify systems and their attributes, and must have mastered a formal systems engineering process model. The arms show the problem being transformed into a solution. The main body consists of the problem solving process which can be applied during System Architecting, System Design and System Project Management. Systems Engineers should take part in all three of these areas. Finally, the feet represent the systems engineering methods and tools that allow systems engineers to effectively and consistently solve problems. Systems engineering methods and tools will change over time as improved ways of problem solving are developed. However, system engineering principles are time tested guidelines that help a systems engineer as they employ new methods and tools. [DeWeck, 2006]

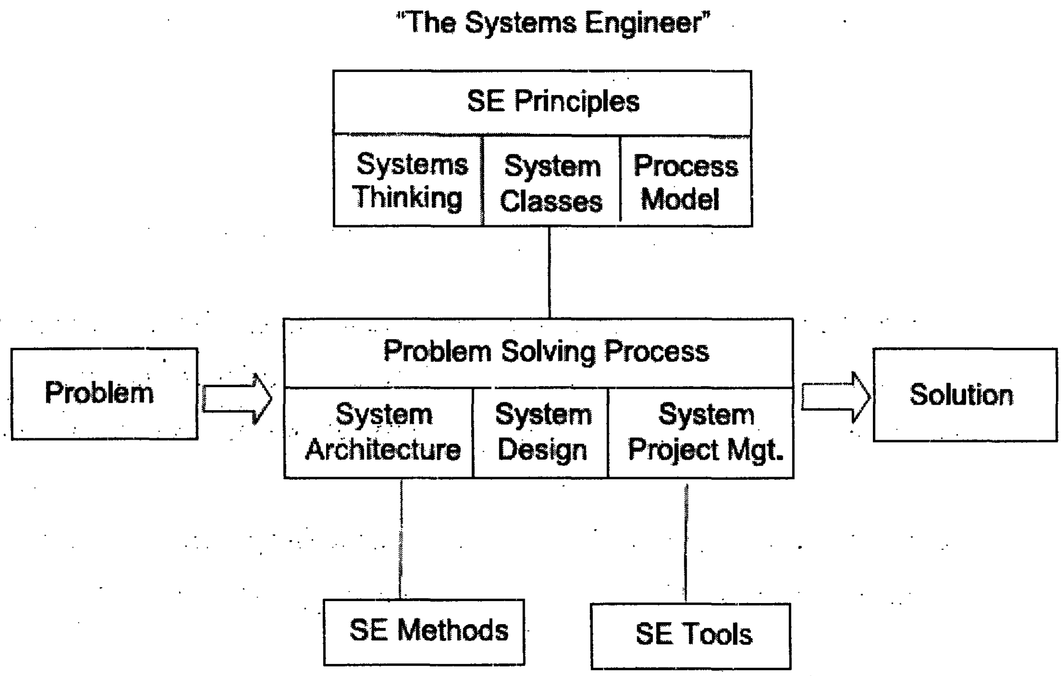


Figure 3.1: Graphical Representation of "The Systems Engineer" [Source: DeWeck, 2006]

This thesis maintains a holistic systems engineering framework. Although the case study of this thesis zooms in on the product development organization and processes at the vehicle subsystem level and examines these processes at a high level of detail, improvements made at this level will be tied back to value delivered to the whole product system [Crawley, 2006]. In this thesis, the whole product system consists of the following:

→ The product/system: Vehicle Design, the Product Development Organization and the Product Development Processes.

→ Customers and Stakeholders (listed in Table 3.1)

→ Supporting Systems (listed in figure 3.2)

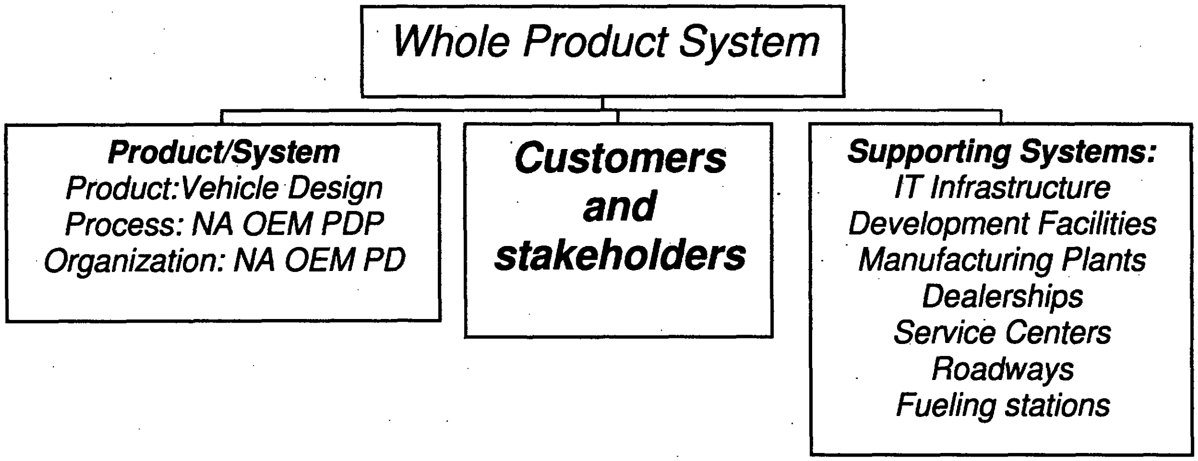


Figure 3.2: The Whole Product System

All decisions made within NA OEM PD organization must create value for the stakeholders of this product system listed in Table 3.1.

Stakeholder Category	Entity
Enterprise	NA OEM
End user	• Manufacturing • Finance • Marketing • Customer Service Center • Dealership • Purchaser/Vehicle Owner/Vehicle User/Passengers
End customer	• Vehicle buyer
Partners	• Suppliers
Suppliers	• Component Supplier • Sub-assembly supplier • IT • HR
Employees	• Product Engineers • Supervisors • Functional Managers • Technical Specialists • Project Managers
Leadership	• PD Management
Society	• Government Regulators
Union	• UAW

Table 3.1: NA OEM PD Stakeholders

This thesis is not focused on improving a subsystem design as a single static event. Rather, this thesis focuses on product development as a dynamic process that involves many organizations within the enterprise and affects many stakeholders both internal and external to the enterprise. As new vehicles are conceived, designed and launched, the product development process and organization should always be evolving as lessons are learned from one product development effort to the next.

As stated, a systems engineering approach uses specific systems engineering principles, methods and tools and can help an OEM's product development organization and processes achieve product designs that meet customer defined attribute targets and are delivered "on time." Systems engineering methods such as the Design Structure Matrix, discussed in Chapter Five, can help enterprises manage complex products such as a vehicle system. Product development projects, such as vehicle programs, involve a complex set of activities that may require coordinating the work of thousands of engineers, managers, technicians, and other professionals in several companies. The work of any one design task can affect many other development decisions throughout the organization [Smith & Eppinger, 1997]. The DSM systems engineering method enables an enterprise or organization to understand and better manage this complex set of activities allowing vehicles to be designed to higher quality standards and within shorter cycle times.

To promote a deeper understanding of systems engineering, a relatively new field in engineering, several useful definitions of systems engineering are listed below:

"Systems engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation... Systems engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."

- The International Council on Systems Engineering (2006)

"Systems engineering is a branch of engineering that concentrates on the design and application of the whole as distinct from the parts...looking at a problem in its entirety, taking into account all the facets and all the variables, and relating the social to the technological aspects."

- Ramo, cited by Clausing, Cohen, & Phadke (2002)

"A systems approach is one that focuses on the system as a whole, particularly when making value judgements (what is required) and design decisions (what is feasible). At the most fundamental level, systems are collections of different things which together produce results unachievable by the elements alone."

- Rechin & Maier, The Art of Systems Architecting, 2002

"Systems exist to satisfy needs, and the complexity arises because only the cooperation of the different elements of the system can yield this satisfaction. If you could partition the system so that each little piece satisfied one of the needs then you would not need system engineering."

- Daniel Whitney, 2007

3.2 Systems Engineering at NA OEM: Systems Focus vs. Components Focus

Although the New PDS at NA OEM has made some improvements in shifting the engineering focus from subsystem/component engineering to systems engineering, most engineering efforts are still focused on the optimization of subsystem and component attributes. Quality and cost reduction efforts are predominantly done at the subsystem and component level. Systems engineers at NA OEM are rarely required to deliver system level quality improvements and cost reductions. This focus is due to the fact that it is easier to make changes at the subsystem/component level. System level changes usually are architectural in nature and require more advanced planning than usually allowed by management for quality and cost improvements. Management often requires instantaneous cost reduction and quality improvement results that meet year end objectives. System level changes are not made as easily as component level changes. Strategies for system level changes must be devised at the very beginning of a vehicle program. Indeed quality and cost changes can be made at the system level. However, these changes require early planning and management support. These cost and quality changes will be generated by changing components and subsystems but these changes have to be coordinated, hence the need to take a higher system-level view.

In addition, the subsystem/component focus at NA OEM is due to the difficulty of cross organizational communication. The NA OEM PD organization has become more functionally oriented. Engineers of the same subsystems/components sit together to promote communication and expertise of individual subsystems/components. However, there is little effort to facilitate cross organizational communication between two engineers who share an interface or whose design parameters both contribute to single vehicle level attribute.

The New PDS at NA OEM does promote cross organizational communication at various points along the product development cycle. For example, early in the program teams of component design and release engineers, purchasing people and suppliers are required to meet to set component pricing goals and to devise a road map to meet these goals. Later in the program a team comprised of the component design and release engineers, purchasing, material planning and logistics, and the supplier meet to ensure that robust manufacturing plans are in place for launch of the component design. However, these cross organizational teams are still widely component focused.

Another indicator of a component focus at NA OEM is the PD organization's efforts to create Product Design Rules. Component and Subsystem engineers are required to write and adhere to design rules. However, these design rules are focused at the component/subsystem level and it is difficult to find many system level design rules, such as design rules standardizing robust interfaces between mating components/subsystems. Design rules are kept at the subsystem level and there is no formal system of vehicle level system integration design rules [Noor, 2007].

One indicator of a weakness in a product development organization's systems engineering capability is the number of system level attributes that fail to meet customer expectations. Often there is not a clear understanding of how various subsystem design parameters interact and affect vehicle level attributes. All vehicle subsystems are decomposed into components with component level engineers assigned to design them. However, not all vehicle subsystems have integration engineers assigned to manage the attributes of the entire subsystem and ensure robust integration into the vehicle system. And if subsystem integration engineers do exist, their job responsibilities are not always well defined.

There is also evidence that component engineers are rarely afforded the chance to engineer their components with a holistic view of the entire product system described earlier. Component engineers have little chance to interact with customers and even find it difficult to spend time in the vehicles for which they are designing components. An engineer at the component level states, "I see little focus on the total system. I am unable to trace my component level performance objectives back to the customer. Objectives for performance, cost, weight, and quality are cascaded to me without rationale for how these quantitative figures were derived."

In order to compete in today's automotive market, automotive companies must use a holistic systems engineering approach when examining their product development processes. Systems engineering principles, methods and tools can enable an automotive OEM's product development organization to create new product designs with attributes that meet customer expectation and deliver these new products to the market in less time than before. Automotive OEMs should not merely focus on product improvement. They need to focus on process and organizational improvements which will ultimately yield improved product quality. The Design Structure Matrix systems engineering method is a way to create an abstraction of current product development processes which can be studied and facilitate process and organizational improvement ultimately leading to product improvement.

3.3 Vehicle Attributes

Merriam-Webster dictionary definition of an "Attribute:"

1 : an inherent characteristic; *also* **:** an accidental quality

Both definitions apply to our discussion on Vehicle System level Attributes. Attributes are inherent characteristics of the vehicle system. However, unexpected characteristics can be thought of as an "accidental quality" of the system. Inherent characteristics are intended and engineers try to create systems that exhibit them. On the other hand, accidental attributes are unintended and engineers try to avoid them.

All of the vehicle's interfacing subsystems are designed to meet an intended vehicle function. However, it is the specific subsystem designs chosen to achieve the required functions that create the vehicle level attributes. These attributes are considered either negative or positive from the standpoint of the customer, with negative attributes commonly referred to as error states. Many different designs can achieve the same function. However, it is the resultant attributes of that design that make the product desirable or undesirable to the intended customer. For example, a driveshaft's function is to transfer torque though an articulation angle while telescoping to accommodate changes in axial length due to suspension movement. Many driveshaft designs can achieve this function. However, if the driveshaft's unbalance causes the vehicle to vibrate at high speeds, this subsystem design has contributed to a negative vehicle attribute or error state. The goal is not to maximize or minimize vehicle attributes because attributes of complex systems are usually highly coupled. Rather, the goal is to employ systems engineering thinking along with appropriate systems engineering methods and tools to prioritize and trade-off functionality and attributes.

Figure 3.3 is an example of attribute coupling. Here we see that vehicle level attributes are a function of subsystem design parameters. The vehicle level attributes are in boxes. The design parameters are in circles. These parameters are grouped around subsystems with names in the respective central circles. Attributes are coupled by common dependent subsystem design parameters. Figure 3.3 shows how the vehicle level attribute 2nd order NVH is coupled with ride quality and vehicle weight, two other vehicle level attributes. Several cross functional subsystems control 2nd order NVH through design parameters. Many of these subsystem design parameters affect both 2nd order NVH and ride quality. Also, many design parameters affect both 2nd order NVH and vehicle weight. It is also interesting to note that, in figure 3.3, two-way arrows represent a feedback loop during the design phase that occurs between the vehicle level attribute and the subsystem design. Whereas, a one way arrow represents the design parameter as an input only to the vehicle level attribute. This means that the design parameter cannot be used as a knob to change that attribute's response. For example, the "axle seat angle" design parameter affects second order NVH. The two-way arrow represents the fact that "axle seat angle" can be varied to achieve a desired 2nd order NVH attribute response. In contrast, "axle pinion length" affects second order NVH. However, it is purely an input into 2nd order NVH response and cannot be changed to achieve desired 2nd order vehicle level response. This is because the axle pinion length design parameter is set by a higher priority vehicle level attribute or functional requirement such as maximum vehicle payload, calling for a certain axle pinion size. Chapter seven will discuss how incorrect assumptions on which design parameters can be controlled to achieve attribute targets artificially limit the design space.

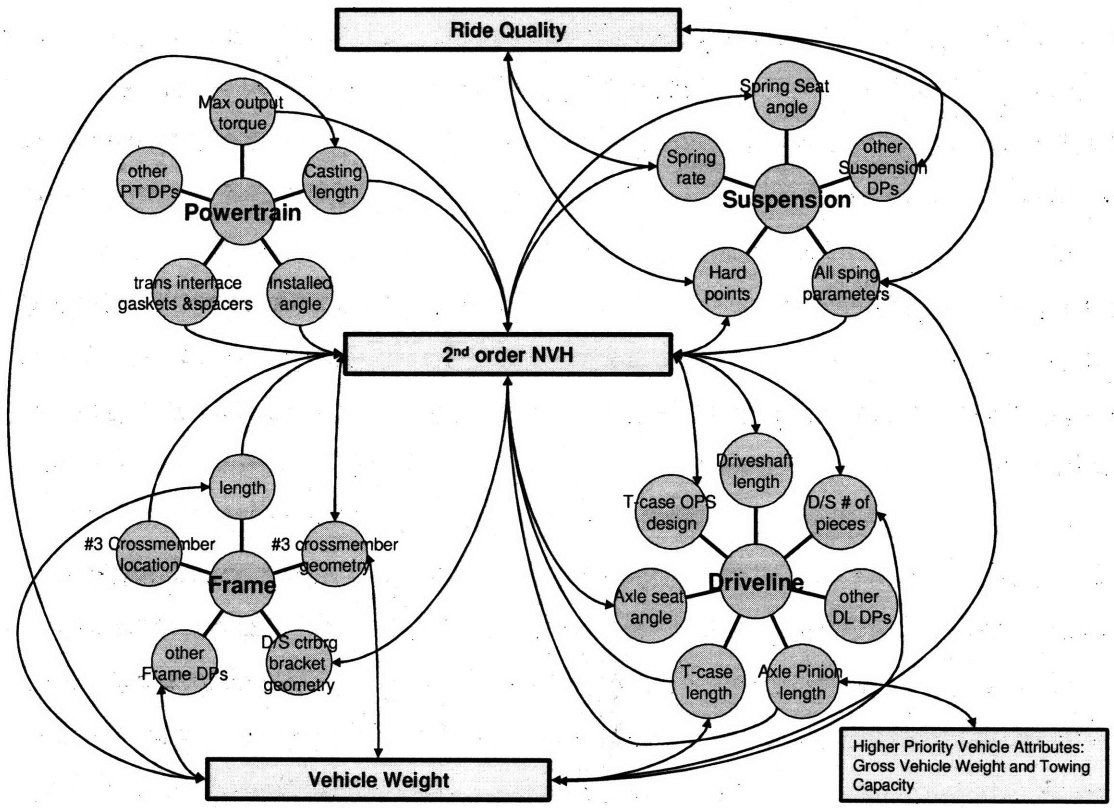


Figure 3.3: Coupling of 2nd order NVH with Ride Quality and Vehicle Weight Attributes

Translation of Customer Needs into System Level Attribute Targets

The Product development process must facilitate successful translation of customer needs into system level attribute targets.[2] Customer needs are often *qualitative* statements, such as "The car should be quiet." Attribute targets are a *quantitative* statement of exactly what the system must achieve in order to meet customer needs and wants.[3] The product development process must also facilitate the successful decomposition and cascading of vehicle level attribute targets to subsystem level attribute targets. Note that attribute targets differ from design parameters or

[2] It is important to note that organizations may also refer to attributes as system characteristics.
[3] Note that various PD organizations may refer to attribute targets as product specifications, product requirements, product objectives or engineering characteristics.

design variables. Attribute targets tell a PD team what to design. Whereas, design parameters specify how to design the system. [Ulrich & Eppinger, 2004] For example, a design parameter might be the required rear axle ring gear surface finish grind specification. In contrast, an example of an attribute target or specification is maximum sound level of axle whine measured at the driver's outboard ear and specified in decibels. In this example, the design parameter "Rear axle ring gear surface finish grind" directly affects the vehicle level attribute "sound level at the driver's outboard ear." The translation of customer needs into engineering system specifications is accomplished by specific methods and tools such as the House of Quality [Hauser & Clausing, 1988].

Faulty vehicle level attribute targets can cause issues all throughout the product development timeline. Early on, issues can occur when vehicle level attribute targets are not realistically linked to customer need. For example, in the case of 2nd order NVH, vibration level targets may be set for a specific driving condition which are too stringent and drive unnecessary development time and hardware costs into the vehicle. These inaccurate 2nd order NVH vibration level targets may not have been correlated with levels of vibration that disturb a customer representing a given percentile driver.

Also, early in the PD process vehicle level attributes targets may not be translated into subsystem level targets in a meaningful or rational way. For example, a vehicle level target for total first order Driveline NVH exists. The first order Driveline NVH target must be decomposed and cascaded to all of the driveline subsystems that contribute to 1st order NVH. It would be in error to cascade a more stringent driveshaft imbalance specification than the rear axle imbalance specification because the two imbalances work to offset each other. If a lower driveshaft imbalance target exists, then the total system effect would be worse 1st order NVH than if equal imbalance specifications are cascaded to the driveshaft and axle subsystems. Vehicle level target decomposition and cascade is discussed in more detail in chapter seven.

Issues can also occur when attribute targets are cascaded down from the vehicle level to the subsystem level without any feedback and negotiation. The process of target cascade is discussed in more detail in chapter seven. Also, issues can occur when there is a mismatch between who is responsible for ensuring attribute targets are met vs. who owns the design parameters that control that vehicle attribute. Attribute ownership and control are discussed in detail in chapter seven as well.

Later in the program, problems can arise when vehicle attribute targets are ill defined. For example, many of the 2nd order NVH attribute targets are described in terms of a subjective rating scale where rating can vary from one engineer to another. Thus, this attribute target is not well defined and should be redefined in terms of an objective measurable target.

Finally, at product launch, issues will occur when the vehicle attribute targets are not met and the product is delayed. Thus the process of vehicle level attribute target setting as well as the process of decomposing and cascading to subsystems and components is a critical process in any organization's PD process.

Prioritization of Attributes

Because vehicle attributes are coupled and design space is limited, attributes must be prioritized. Attribute coupling was defined in section 3.3 and depicted in Figure 3.3. The case study for this thesis focuses on a vehicle attribute known as second order noise vibration and harshness (2nd order NVH). This attribute will be described in more detail in chapter four. Although this thesis is focused on improving 2nd order NVH, it is not focused on optimizing this attribute at the expense of other attributes that could be more important to the customer. Rather, the PD organization must know the prioritization of vehicle attributes and know when trade-offs between, for example, vehicle dynamics performance and 2nd order NVH, can be made. Or when trade-offs between vehicle weight, tied directly to fuel economy, and 2nd order NVH can be made. Discussed further in chapter seven, NA OEM tackles the issues of attribute prioritization is by creating "brand attribute DNA" and program attribute leadership strategies. Based on target customers, the PD organization tailors attribute targets for each vehicle program. Then, attributes within a program are prioritized based on the desired level of competitiveness with other benchmark vehicles. It is essential that these prioritizations are cascaded in a meaningful way to the vehicle subsystem and component levels. Subsystem and component engineers must know when to trade off weight and cost and other attributes targets cascaded to them as they explore their design space throughout the product development process. They must know the answer to "Which attribute is more important to the customer, the enterprise and all stakeholders?" and should understand why this prioritization exists.

Integration of Subsystems

Attributes are emergent properties of a whole integrated system [Loureiro, Leaney & Hodgson, 2003]. In order for a system design to meet the attribute targets derived from customer needs, all subsystems must be designed and integrated such that these system level attribute targets are met. Often several different subsystems or in some cases, all of the subsystems, contribute to a vehicle attribute.

Subsystem interface control is critical when subsystems are being designed and integrated into the total system. In order for subsystems to be integrated into the total system, such that the system functions as intended under all conditions, all the subsystem interfaces must be well understood and managed. Subsystem interfaces include mechanical interfaces, spatial interfaces, energy flow, material flow and information flow. Subsystem interfaces often represent PD cross organizational interfaces. The PD organization is based on a simple hierarchical decomposition of the vehicle systems. However, the subsystems that are integrated to form the total system have many interactions amongst each other that cannot be represented by this simple hierarchical or modular decomposition. Therefore, there are many cross organizational design problems that must be resolved. Some questions that arise are the following: How are subsystem interfaces designed and controlled? How is cross-organizational communication managed and its completeness ensured? How are cross organizational design interface decisions made? How are system level "lessons learned" documented and accessed? Figure 3.4 demonstrates the cross organization nature of engineering the vehicle subsystems such that the 2nd order NVH attribute targets are met. This figure represents the cross organization engineering collaboration that is required to deliver second order NVH vehicle level requirements. This required cross organizational design collaboration is the case for most vehicle level attributes.

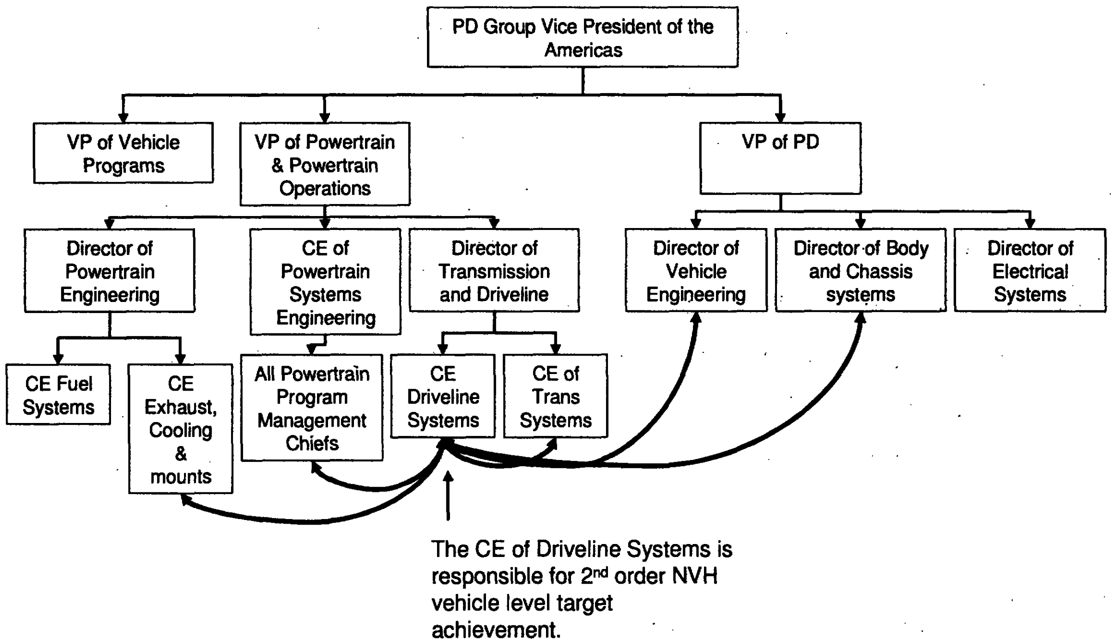


Figure 3.4: Cross Organizational Communication Required for 2nd Order NVH Attribute Management

Ownership and Control

As stated previously, vehicle attributes emerge when all of the subsystems are integrated. Thus, the question emerges; who is responsible for ensuring the vehicle meets system level attributes targets? At the same time, who controls the design parameters that affect each vehicle attribute? Often, these are not the same product development teams and sometimes they are not even in the same management reporting chain. At NA OEM, we find ownership of some vehicle level attributes at the subsystem level. 2nd order NVH, the case study discussed in this thesis, is one example of a vehicle level attribute that is owned at the subsystem level. As explained in chapter four, the driveline subsystem organization is responsible for signing off vehicle level testing that states that 2nd order NVH attribute targets are met. However, as shown in Figure 3.4, the driveline organization does not control all of the design parameters that contribute to 2nd order NVH. Many of the design parameters are controlled by the suspension group within the chassis organization. The powertrain and chassis organizational reporting chains only merge at the very top of the PD organization with the Executive Vice President of North America PD. Driveline system engineers are tasked with ensuring that chassis design parameter inputs to vehicle level 2nd order NVH are well understood. However, communication between the two organizations is not formalized in New PDS. "Good" driveline systems engineers take it upon themselves to communicate with chassis engineering. But, without formalization of this communication, these chassis inputs can "fall through the cracks" and cause problems during vehicle level testing [NA OEM, 2007].

3.4 Conclusion

An introduction to systems engineering and vehicle level attributes are deliberately combined in this chapter to emphasis that the two are intrinsically related. In order for the vehicle system to meet attribute requirements a systems engineering approach must be employed in the product development processes and organization.

4. CASE STUDY: SECOND ORDER NOISE VIBRATION AND HARSHNESS

4.1 NA OEM Product Development Organization

As previously stated, the case study for this thesis focuses on second order NVH vehicle level attribute which is currently owned by the driveline subsystem. This vehicle level attribute serves as a case study for system engineering and vehicle attribute management. Thus, it is important to gain an understanding of the driveline subsystem organization's position within NA OEM's Product Development (PD) organization.

Within the PD organization, the entire vehicle system is divided up into several major subsystem level functional organizations. Figure 4.1 below shows the structure of NA OEM PD organization. Although, presently in 2007, NA OEM PD remains a matrix organization, numerous reorganizations occurring over the past two years have shifted NA OEM PD toward a more functionally oriented matrix organization. Within NA OEM PD's functionally skewed matrix organization, the functional organizations serve to promote a desired level of subsystem level expertise among PD engineers. NA OEM's major subsystem level functional areas are Powertrain, Chassis & Suspension, Body, and Electrical.

Due to the major subsystem level complexity of the powertrain organization, this organization is further broken down into a subsystem level which is comprised of engine, transmission, driveline, and powertrain systems (exhaust, fuel, mounts and cooling). Each of these four Powertrain functional areas is under the jurisdiction of a corresponding functional chief engineer. Within some of these four Powertrain functional areas, there are corresponding subsystem engineering groups that serve to integrate all of the components within each of the four major subsystems. These subsystem engineering groups are tasked with ensuring that their subsystem will meet all attribute targets cascaded by the vehicle engineering organizations. However it is important to note that the target cascade is not just a one way cascade. Targets are proposed to the subsystem engineers and the subsystem engineers are responsible for negotiating the final agreed upon value with vehicle level engineers.

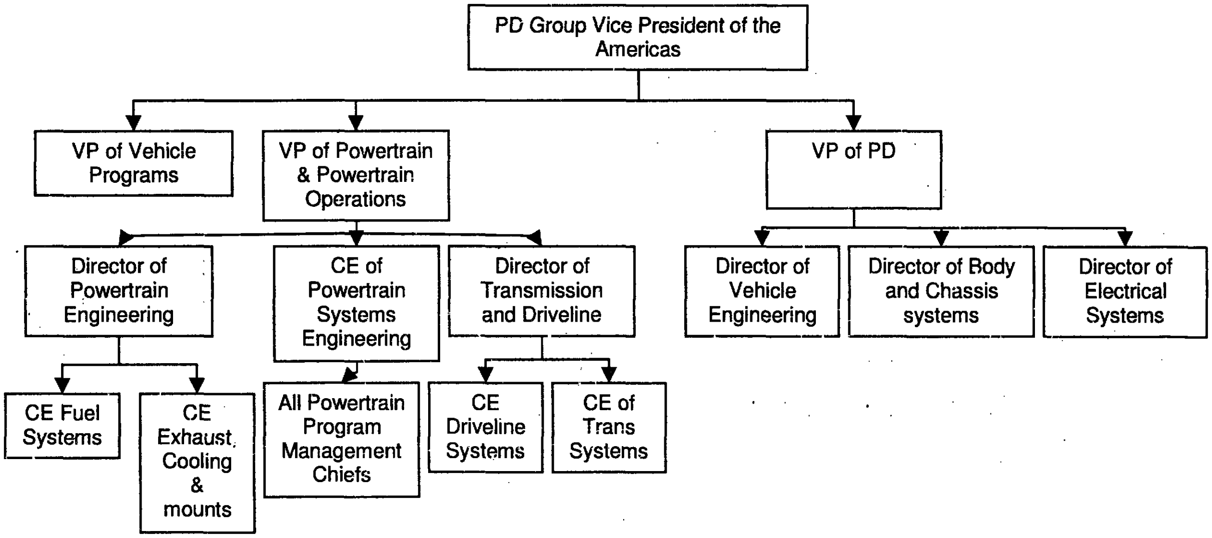


Figure 4.1: Product Development Organization Structure at NA OEM

4.2 The Driveline Subsystem

In order to better understand the technical content of this thesis, this section provides a lesson on the vehicle's driveline subsystem. The driveline's primary function is to transfer powertrain rotational speed and torque to the wheels, enabling the intended response at the interface between the vehicle wheels and the road. Figure 4.2 shows a hierarchical decomposition of the vehicle powertrain subsystem. The powertrain is considered a level one subsystem. As explained in the previous section, the powertrain is further decomposed into several level two subsystems, one of which is the driveline subsystem. The driveline subsystem can then be further decomposed into several level three subsystems.

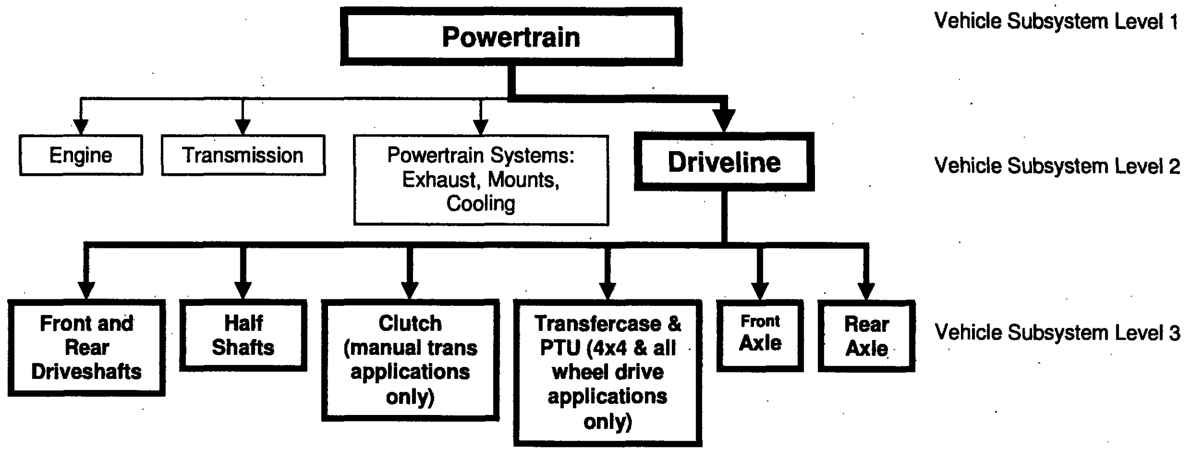


Figure 4.2: Hierarchical Decomposition of the Powertrain Subsystem

The driveline subsystem is comprised of the following level three subsystems: transaxle (for front wheel drive and all wheel drive vehicles only), clutch (for manual transmission applications), transfer case (for 4x4 and AWD applications), front axle (for 4x2 and 4x4 applications), rear axles, front and rear drive shafts, and half shafts (for independent suspension applications). These subsystems interface with most other vehicle subsystems and the interaction between each of these subsystems are factors in determining vehicle level attribute responses. In figure 4.2 the driveline, a vehicle subsystem level 2, is responsible for ensuring that all vehicles meet 2nd order NVH vehicle level targets. Within driveline, the rear driveshaft, transfer case, and rear axle control 2nd order NVH. However, as shown previously in figure 3.4, many subsystems outside of driveline and powertrain control 2nd order NVH vehicle response. It is important to note, at this point in time, that an interface can be defined as physical/mechanical, spatial, energy transfer, material exchange and/or information exchange.

The driveline architecture varies based upon the vehicle level architecture and vehicle level requirements. The most common driveline architectures are represented in figures 4.3 and 4.4. These figures depict how the various driveline subsystems interface with each other physically. Front wheel drive, rear wheel drive, 4x4 and all wheel drive architectures are shown in these figures. The intended functions of all two wheel drive driveline subsystems are to transmit rotation and torque amongst the driveline subsystems, reduce rotational speed and multiply torque and split the torque left-to-right and enable left-to-right wheel speed differences.

The intended functions of all Four Wheel Drive and All Wheel Drive systems include those listed for two wheel drive and also include; split the torque from front to rear, allow front to rear axle speed differences and prevent runaway wheel slip.

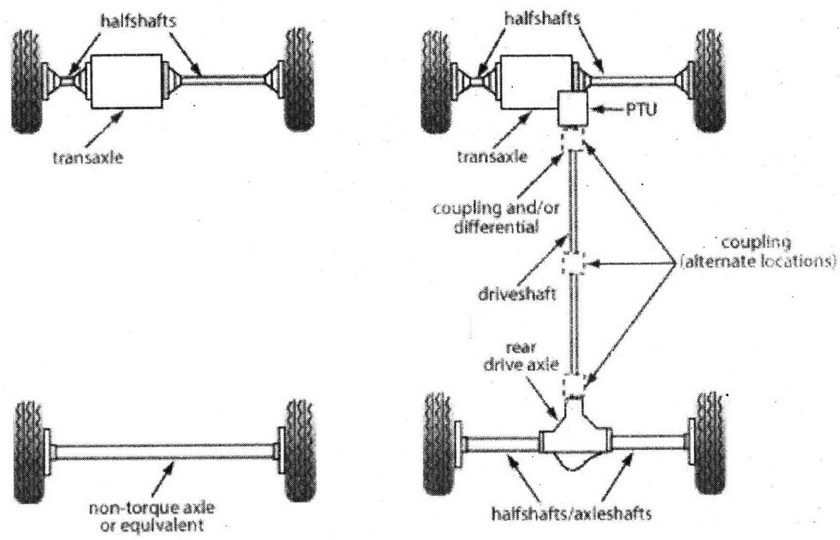


Figure 4.3: Front Wheel Drive architecture (Two Wheel Drive and 4x4/All Wheel Drive)

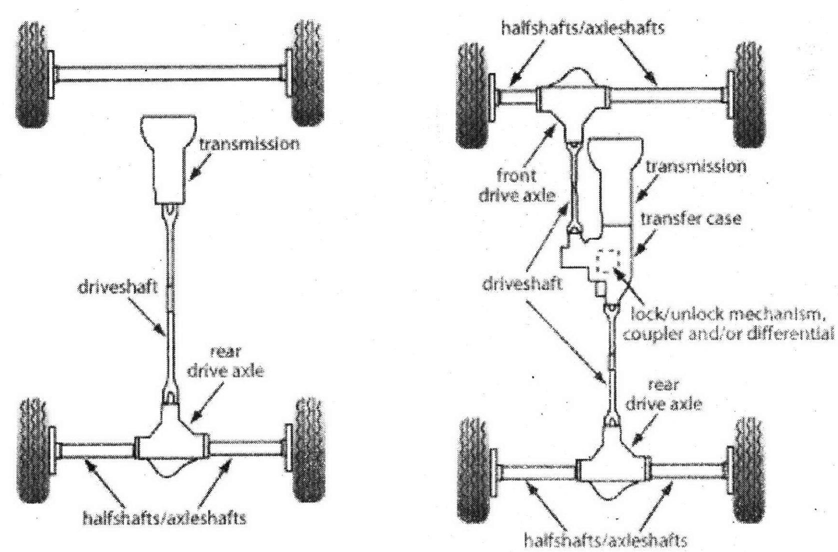


Figure 4.4: Rear Wheel Drive architecture (Two Wheel Drive and 4x4/All Wheel Drive)

In order to convey the interaction between the driveline subsystems, an object based design structure matrix is shown in figure 4.5 which represents the interactions between all of the various driveline subsystems. In figure 4.5 we can see that material flow, in the form of lubricating grease, can exist between the front driveshaft and the transfer case if a slip yoke/stud yoke architecture is chosen to accommodate axial plunging of the front driveshaft. We can also see that information flow can exist between the transmission and the transfer case in the form of powertrain controls. Table 4.1 summarizes the driveline subsystems, their primary functions and vehicle attributes that they affect.

	Front Halfshafts/axleshafts	Front Drive axle	Front Driveshaft	Transmission	Transfer case, power transfer unit and coupling	Rear Driveshaft	Rear drive axle	Rear halfshafts/axleshafts
Front Halfshafts/axleshafts		E,P						
Front Drive axle	E,P		E,P					
Front Driveshaft		E,P			E,M,P			
Transmission					E,M,P,I			
Transfer case, power transfer unit and coupling			E,M,P	E,M,P,I		E,M,P		
Rear Driveshaft					E,M,P		E,P	
Rear drive axle						E,P		E,P
Rear halfshafts/axleshafts							E,P	

E = Energy Flow
I = Information Flow
M = Material Flow
P = Physical Interface

Figure 4.5: Object Based DSM Representing Driveline Subsystem Interfaces

Driveline Subsystem	Primary Function	Vehicle Attributes Effected
Axles (front, rear, transaxles)	• Change direction of the power flow 90°. • Provide gear reduction. • Split torque between left & right wheel. • Allow wheel speed differentiation.	• NVH • environmental resistance • lifetime durability • serviceability
Driveshafts & Halfshafts	• Transmit rotation and torque between driveline components. • Allow angular driveline motion. • Allow for axial driveline plunge and extension.	• NVH • Lifetime durability • Vehicle system packaging • Vehicle assembly • Crashworthiness • Serviceability
Transfer case, Power Transfer Unit (PTU) and Coupling	Transfer case: • Distribute torque between front & rear axles. • Drive rear axle with front axle disconnected. Coupling & Power Transfer Unit: • Control undesirable RPM variation between the front and rear axles. • Enable front and rear axle speed differences while the vehicle is turning. • Send torque to the rear axle. • Decrease torque and increase the RPM	• Transfer case: NVH • Coupling: NVH, handling, traction • Power Transfer Unit: NVH

Table 4.1: Driveline Subsystem Primary Function and Attribute Effect

4.3 Subsystems that Interface with the Driveline Subsystem

Once integrated into the vehicle, the driveline, as defined above, interfaces with almost all other vehicle subsystems. The existence of these numerous and varied interfaces with all other vehicle subsystems means that the driveline subsystem design and integration into the vehicle must be well managed by the product development processes. The object-object design structure matrix in figure 4.6 shows the main driveline components and their interfaces with the other major vehicle subsystems. Physical, spatial, energy, material, and information interfaces are represented.

	Front Halfshafts/axleshafts	Front Drive axle	Front Driveshaft	Transmission	Transfer case, PTU & coupling	Rear Driveshaft	Rear drive axle	Rear halfshafts/axleshafts	Front Suspension	Brakes	Frame	Engine	Exhuast	Mounts	Fuel	Rear Suspension
Front Halfshafts/axleshafts		E,P							E,P	E,P	S					
Front Drive axle	E,P		E,P						E,P		S					
Front Driveshaft		E,P			E,M,P						S	S	E,S			
Transmission					E,M,P,I						S	E,P,I	E,S	E,P	S	
transfer case, PTU & coupling			E,M,P	E,M,P,I		E,M,P					S		E,S	E,P	S	
Rear Driveshaft					E,M,P		E,P				E,P,S		E,S		S	
Rear drive axle						E,P		E,P		P	S		E,S			
Rear halfshafts/axleshafts							E,P						E,S			E,P,S
Front Suspension	E,P	E,P								E,P	P	S	E,S			
Brakes	E,P						P		E,P						S	E,P
Frame	S	S	S	S	S	E,P,S	S		P			S	S,P,E	P,E	S,P	E,P
Engine			S	E,P,I					S		S		S,P,M,E	P,E	S,P,M,E,I	
Exhuast			E,S	E,S	E,S	E,S	E,S	E,S	E,S		S,P,E	S,P,M,E		E,S	S,E	S,E
Mounts				E,P	E,P						P,E	P,E	E,S			
Fuel				S	S	S				S	S,P	S,P,M,E,I	S,E			S,E
Rear Suspension								E,P,I		E,P	E,P		S,E		S,E	

E = Energy Transfer
I = Information Flow
M = Material Flow
P = Physical Interface
S = Spatial - Package Space

Figure 4.6: Object-Object Base DSM Representing Driveline Interface with Other Vehicle Subsystems

4.4 Vehicle level Attributes affected by Driveline Design

The driveline subsystem design affects many vehicle level attributes. The vehicle level attributes affected by driveline design and experienced by the end user are the following:

- Powertrain Noise, Vibration and Harshness (NVH):
    - 1st order vibration felt in the steering column, seat track and floor.
    - 2nd order vibration felt as start up shudder or heard as high speed moan.
    - Radiated noise heard as a whine from the axle or transmission.
    - "Clunk" felt as a jerking motion on acceleration.
    - "Boom" heard as a booming sound in the cab when driving at various speeds.

- Other vibration, for example, from the axle pinion.

- Max allowable vehicle speed
- Ride quality
- Fuel economy (weight & efficiency)
- Manufacturability
- Ease of installation at the final assembly plant (ergonomics)
- Serviceability (ergonomics)
- Gear shift quality
- Cost
- Safety
- Durability

These vehicle attributes, affected by driveline design parameters, are coupled. For example, driveline design parameters that improve fuel economy may also improve 1st order NVH and cost. A single piece driveshaft weighs significantly less than a two piece driveshaft. At the same time a single piece driveshaft can significantly improve 1st order NVH response due to the absence of a center bearing assembly. On a multi-piece driveline, the center bearing assembly acts as a direct path for 1st order vibration created by driveshaft imbalance to reach the frame, body, and subsequently customer seat. A single piece driveshaft eliminates this sensitive transfer path. Moreover, cost for most single piece driveshafts is lower than two piece driveshafts due to additional hardware required in a two piece driveshaft. Therefore, some auto manufacturers have developed a common architecture for all trucks where ample underbody package space has been created to enable single piece driveshafts for all truck applications. There are numerous other couplings amongst these driveline design affected vehicle attributes which means that trade-offs between attributes must be carefully considered when finalizing the driveline subsystem design parameters.

4.5 Second Order Noise, Vibration and Harshness

As a case study for how "high impact" vehicle attributes can be better managed within an NA OEM's functionally divided organization, this thesis will focus on the vehicle level attribute labeled as second order noise, vibration and harshness (2nd order NVH). 2nd order NVH can be considered a high impact vehicle attribute. For the sake of this thesis, high impact vehicle attributes will be defined as those that occur on vehicles produced at high volume and that are either historically problematic at launch or occur as one of the top ten issues in a warranty R/1000 pareto. R/1000 is the number of warranty claims per 1000 vehicles produced.

This thesis focuses on the vehicle level attribute 2nd order NVH because it can be characterized as a "high impact" vehicle level attribute. 2nd order NVH historically plagues NA OEM truck programs due to large suspension travel, heavy payloads and high torque. Therefore it affects NA OEM's top selling vehicles. It affects vehicle models that sell upwards of 500,000 units per year. Warranty analysis for driveline shows 2nd order NVH as a top 10 issue for some vehicle models. In addition this vehicle attribute has historically caused issues during vehicle prototype testing and at vehicle launch.

A Pareto chart of warranty claims from a major truck program at NA OEM was used to generate the pie charts summarizing total warranty claims for a single calendar year. These pie charts, in figure 4.7, show that 2nd order NVH is amongst the top ten warranty issues. In addition, this warranty issue is costly to fix. The fix is labor intensive as dealership service centers must install shims at the rear axle pinion and suspension interface or at the frame and driveshaft center bearing interface.

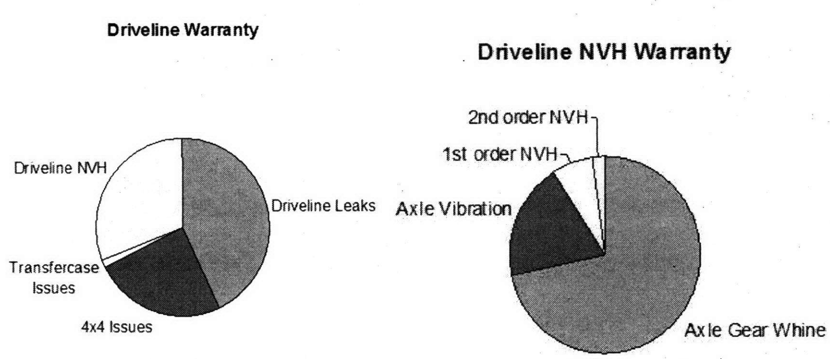
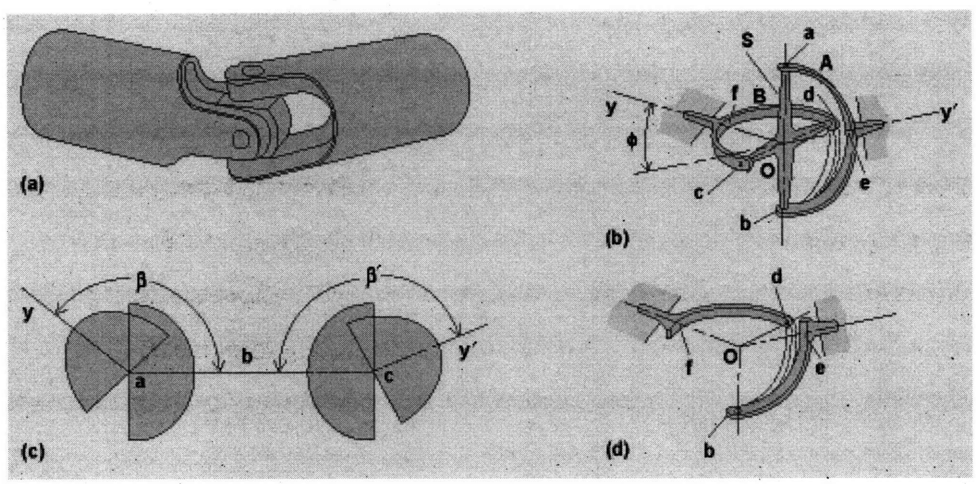


Figure 4.7: Pie Charts Summarizing Total Driveline and Driveline NVH Warranty

In addition, 2^{nd} order NVH issues have notoriously emerged at or just before vehicle launch for several predominant truck models over the past few years. The root cause of these 2^{nd} order issues varies as this is a vehicle attribute that has design parameters controlled by multiple subsystems. An interview with a driveline systems engineer working on a high profile truck program shared the following story: Driveshaft component engineer and the driveline systems engineer worked together over a period of four weeks to develop a driveline angle strategy that would yield optimal 2^{nd} order NVH response for all driving conditions. However, at the final verification prototype phase, all of the prototype vehicles built had much lower rear axle pinion angles than the suspension engineering team predicted. The unintended driveline angles caused severe shudder during vehicle take off from a stop, a 2^{nd} order attribute error state. Thus, the driveshaft design and release engineer and the driveline systems engineer had to rework the entire driveline angle strategy. This late design issue is just one example of many 2^{nd} order NVH vehicle level issues that have occurred at or just before launch.

The 2^{nd} order NVH vehicle attribute was also chosen as the case study for this thesis because of its cross-functional and cross-organizational nature. While the driveline organization is responsible for ensuring that the vehicle level attribute targets for 2^{nd} order NVH are met, many of the design parameters and associated manufacturing variation are controlled by other engineering organizations within NA OEM's PD. Therefore, issues arise when these organizations make changes to design parameters that affect 2^{nd} order NVH without communicating with the driveline organization. Many engineers within other organizations are not aware that changes to design parameters they control affect 2^{nd} order NVH. Thus, this vehicle level attribute will serve as a useful case study allowing the reader of this thesis to gain an understanding of the practical application of systems engineering principles, methods and tools. In addition, the NA OEM PD organization will benefit from improved management of this high impact vehicle level attribute.

The source of vehicle level second order NVH is the driveshaft's single cardan universal joints (u-joint). Figure 4.8 shows a u-joint. A u-joint functions to allow the driveshaft to transfer torque from the powertrain to the axle as the driveshaft rotates through the driveline operating angle called ϕ in Figure 4.8. A u-joint consists of two yokes attached to their respective shafts and connected by a spider enabling a u-joint angle. For automotive applications, ϕ rarely exceeds four or five degrees. Due to the u-joint angle ϕ, the instantaneous angular displacement of the two shafts is only the same every 90 degrees per rotation. This variation in instantaneous angular displacement of the two shafts is the source of 2nd order NVH. In order to minimize 2nd order NVH, an intermediate shaft must be placed between two u-joints and the angles of each u-joint must be approximately equivalent. Thus, for automotive applications, the driveline angles must be carefully planned and controlled. This is a difficult challenge for automotive applications because the driveshaft angles are continually changing due to various suspension positions under various load and driving condition.



a) Simple Single Cardan Universal Joint
b)The Yoke and Spider
c)Double Cardan Universal Joint
d)Four-bar conic linkage equivalent of yoke and spider

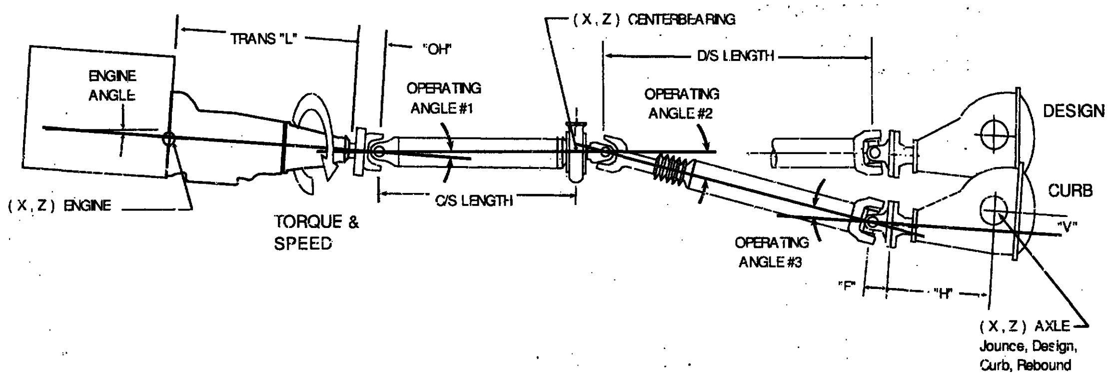
Figure 4.8: Driveshaft single cardan universal joint [Source: Adams, 2007]

2nd order NVH is a function of the following driveline design parameters; max driveline speed, driveline torque, driveshaft and half shaft lengths and driveline operating angles (the angle called ϕ). However, this vehicle attribute is also affected by the following list of design parameters which are controlled by subsystems outside of the driveline engineering organization:

- Body style and feature package variations that affect vehicle weight and ride height
- Suspension design

- Frame design
- vehicle max speed
- Powertrain and transmission torque, length and installed angle
- Engine, trans and body mount rates and architecture

Figure 4.9 depicts a two-piece driveline system and the design parameters that influence 2nd order NVH. These parameters are not all static. Changes in the suspension as it travels from jounce to rebound and as the vehicle is loaded from empty (curb weight) to full gross vehicle weight (design weight) change the driveline operating angles and affect the 2nd order response.



OH = U-joint Overhang
Trans "L" = Transmission length
C/S = Couple Shaft of the two piece driveshaft system
D/S = Driveshaft
X,Y,Z = Vehicle Coordinates
"F" = Ujoint center to flange end length
"H" = Rear axle pinion length from axle centerline to pinion flange end
"V" = height from center of pinion flange to centerline of axle

Figure 4.9: Driveline system and design parameters affecting 2nd order NVH [Source: NA OEM, 2007]

Drivers and passengers of vehicles that have objectionable levels of 2nd order NVH experience a shuddering feeling when they accelerate from a stop. This is caused by dynamic secondary couple forces at the transmission yoke, driveshaft center bearing and axle pinion yoke. Secondary couple forces are a reaction to torque, angles, and lengths that produce a 2nd order shaking force on the yokes and center bearing [NA OEM, 2007]. At higher vehicle speeds, drivers and passengers can also hear a moan while maintaining a constant speed, accelerating or coasting. This moan is caused by torsional accelerations cause by the angular lag and lead associated with u-joints. High operating angles aggravate this condition. The magnitude of the objectionable vibration felt and noise heard by the customer is influenced by customer driving conditions. Do they always drive around with the vehicle unloaded? Or, do they load their vehicles up? Do they pull a trailer? What is the driving speed? How are they accelerating?

The customer driving conditions greatly affect the 2^{nd} order response of the vehicle. The largest customer influenced factor is the total vehicle weight which controls ride height in all driving modes and pinion angle under high torque driving modes. Ride height sets the driveline angle which affects the 2^{nd} order response. Vehicle weight also affects how much the pinion "winds up" under high torque applications. The driveline systems engineer is responsible for ensuring that 2^{nd} order attribute targets are met in all vehicle load and torque conditions. 2^{nd} order NVH targets are specified in terms of a minimum subjective rating. All driveline NVH test engineers must agree that that magnitude of vibration felt during acceleration and sound level of moan at high speeds is low enough to be granted at least the minimum acceptable subjective rating at defined by the customer. 2^{nd} order NVH cannot be eliminated entirely but rather must meet customer defined maximum acceptable levels.

Aside from subsystem design, manufacturing variation also plays a role in 2^{nd} order NVH. Thus, the nominal design parameter value and associated tolerance as well as capability to achieve these values must be carefully managed. Inherent manufacturing variation in the frame, powertrain, transmission, suspension and body mounts will affect a vehicle's 2^{nd} order response.

4.6 Chapter Summary

The driveline organization is a subsystem level two organization that resides with in the product development powertrain subsystem level one organization. The driveline organization is responsible for ensuring that the vehicle level design meets 2^{nd} order NVH targets. These targets are subjective rating levels for vibration and noise magnitude. However, many other subsystems located outside of powertrain own design parameters that control the 2^{nd} order vehicle response. Thus, it is the responsibility of the driveline engineers to coordinate communication with these other subsystems. The driveline subsystem consists of several level three subsystems which are front and rear driveshafts, transfer case/PTU and coupling, front axle, and rear axle.

2nd order NVH is determined to be a high impact vehicle level attribute as it is an attribute that has caused issues at launch and in the field on high production volume vehicles. 2nd order NVH is a result of the driveshaft single cardan universal joint. The forces created when the universal joint rotates through an angle cause 2nd order NVH.

5. INTRODUCTION TO THE DESIGN STRUCTURE MATRIX SYSTEMS ENGINEERING METHOD

5.1 Introduction to the Design Structure Matrix

As discussed, the automotive system can be considered a complex system where thousands of components are integrated into hundreds of subsystems which are integrated into the final vehicle system. These subsystems and components have many interfaces and interactions at the vehicle system level. In addition, the automotive product development process involves thousands of engineers, managers, technicians, and other professionals. The product development processes required to deliver a high quality final vehicle system that meets customer expectations are complex and need to be carefully managed. A systems engineering approach to vehicle attribute management at automotive OEMs can provide a competitive advantage by enabling high quality vehicle designs that meet the desired accelerated product development timing. [Eppinger & Smith, 1997]

One systems engineering method that has been developed in academia and is being adopted in industry to help manage extremely complex products, processes and organizations is the Design Structure Matrix (DSM). Today, the DSM is being widely adopted by corporations that have realized the competitive advantage that this complexity management systems engineering method can provide. A number of corporations are using the Design Structure Matrix to manage complex automotive components systems and subsystems, aerospace configuration design, concept development and program roll-out, electronics and semi-conductor development, equipment and machine tool development, and plant engineering and construction projects [Browning, 2001]. The DSM modeling approach can help to address problems facing complex product development projects within any corporation. In fact, there is an annual "International Design Structure Matrix Conference" held as a forum for academics and corporations to share the latest DSM complexity management developments, trends and ideas [dsmweb.org].

The DSM can be used to analyze and manage system, process and organization design. Therefore, the DSM can be used as both an engineering system method, studying complex interactions in a system design, and a system project management method, studying complex process flows or communication patterns within an organization. When using the DSM as a system analysis method, complex interactions and interfaces between subsystems and components are captured and clearly represented in a compact square matrix. One only needs to know how to read a DSM to know how subsystems interface and interact in an integrated system. When using the DSM as a Project Management method, processes or organizations can be represented. One can see process task interdependencies and understand the flow of information through the process with both feed forward and feedback information flows represented. Or, one can see how individuals within organizations communicate. In this thesis, we will focus on the DSM as a project management method to streamline the vehicle product development process and deliver a better vehicle system. Specifically, the DSM will be used to study subsystem design and integration processes at a high level of detail so that specific vehicle attributes that greatly impact the customer's perception of the vehicle system meet customer defined target values and that these vehicle level attribute targets are met "on time" in an accelerated product development cycle plan. The DSM method can accomplish this by allowing an organization to better document existing procedures, reduce complexity, share data, facilitate project flow, reveal constraints and conflicts and design iteration strategically. [dsmweb.org]

In 1980, Steward introduced the concept of using a square matrix to show dependencies between inputs and output tasks in a process. He created an n^2 matrix with n process tasks in rows and the same n items represented across the top as columns, and placed a dependency mark at the intersection of two tasks with a specific dependency. The DSM provided an advantage over other project management process modeling methods such as PERT and Critical Path Method (CMP) because unlike these two methods, the DSM model can represent the natural iterations that are inherent in any complex product development process [Steward, 1991]. Design iteration is the repetition of a design task due to the appearance of new information. A task may have to iterate for several reasons. First, an upstream task may have to repeat itself when a downstream task discovers that the work done in that upstream task was wrong or incompatible. Second, information in an upstream task that eventually feeds a downstream dependent task may be changed due to a late management decision or may have been in error. Thus, downstream tasks need to be repeated.[Eppinger & Smith, 1997].

Some advantages of DSM as a project management method over other project management methods are the following [dsmweb.org]:

- Compactly diagrams information flow of complex processes and is easy to read and interpret.
- Impact of management decisions on process completion timing can be traced.
- Can be a consensus document for a cross organizational project team.
- Can help all members of a cross functional team see the big picture.

5.2 Types of Design Structure Matrices

There are four different types of DSMs. The four different types of DSMs and a brief description are shown in table 5.1.

Component Based DSM	• Shows interactions between elements in a complex system architecture. • The type of interaction, such as spatial, physical, material flow, information flow, and energy flow, can be represented.
Team Based DSM	• Used to analyze an organization and is based on information flow among individuals and groups working on a specific project. • Is created by identifying communication flows and representing them as connections between individuals and groups in the matrix. • The type of information flow and its frequency should be represented.
Activity Based or Task Based DSM	• All tasks in a process are represented and information flow between tasks is represented by a mark at the corner of the interaction. • Task inputs and outputs are represented. • Parallel or concurrent, sequential or dependent, independent, and iterative task relationships are represented. • Information feed forward and feedback are represented.
Parameter Based DSM	• Used to analyze system architecture based on design parameter interrelationships. • Created by explicit definition of a system's decomposed design parameters and their interactions. • The types of interactions among system design parameters should be indicated as well as the associated strength of the interaction.

Table 5.1: Four Types of Design Structure Matrixes

An activity based or task based DSM will be used in this thesis to map the current process of vehicle level attribute target cascade to subsystems, setting subsystem design parameters, integrating the subsystems and testing at the vehicle level such that the vehicle level attribute known as "second order noise, vibration and harshness" (2^{nd} order NVH) meets target values. An activity based DSM is chosen because a careful analysis of the current process tasks and their interdependencies and iterations will enable process improvement to be discovered. Figure 5.1 shows that two of the DSM types are static and two are time-based. A time-based DSM tracks the flow of information over time. Explained later in this chapter, with data on task completion time and probability of rework, an activity based DSM can produce stochastic process lead time predictions. The activity based DSM used in this thesis case study is time based and process data collected will be used to simulate total process time for the "As-Is" and "To-Be" product development process.

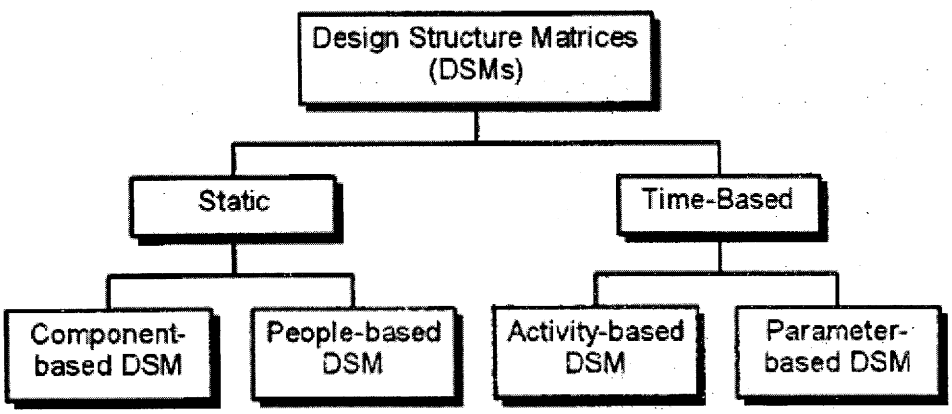


Figure 5.1: Static vs. Time Based DSM

5.3 Understanding the Binary Activity-Based DSM

In an *nxn* square matrix such as the Activity-Based DSM, *n* number of tasks comprise the process under examination and are listed in both the row and column headers. A binary Design Structure Matrix is one that shows a task dependency with a mark of unity, "1," and no dependency with a mark of zero, "0," or no mark at the intersection of two tasks. Dependency between two tasks could also be marked with an "X." The tasks listed in the column headers are considered inputs to the tasks listed in the row headers. Therefore, if you read across a single row in the DSM, you will discover all of the tasks that are required as inputs to that single task. If you read down a single column of the DSM you will discover all of the tasks that that particular task feeds. In other words, you discover what tasks are dependent on the information created by that single task. Two tasks or two groups of tasks can have several types of relationships with each other. These can be summarized and dependent/sequential/series, independent/parallel, and interdependent/coupled. Examples of these task relationships between two tasks or two groups of tasks, A and B, are shown in figure 5.2.

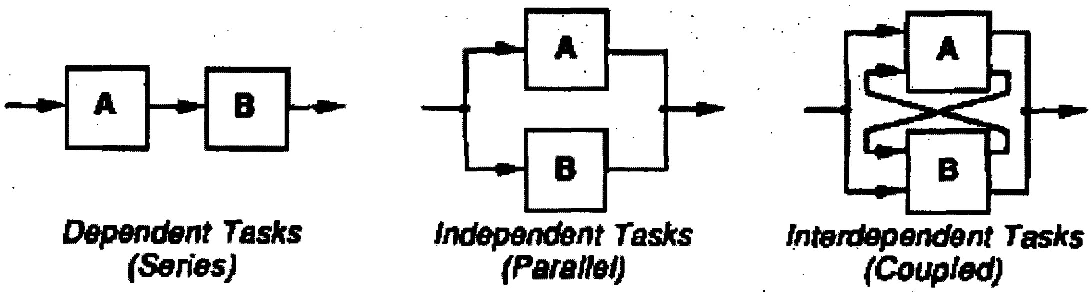


Figure 5.2: DSM Task Dependency Types [Source: Eppinger, 1991]

Figure 5.3 is an example of a binary activity based DSM and represents the process of designing the transmission to driveshaft interface. Here we see that tasks A, "Set vehicle level targets," and B, "set driveshaft subsystem targets" are dependent and thus must be executed in series. Tasks A and C, "set transmission subsystem targets" are also dependent. We can see that tasks B "set driveshaft subsystem targets" and task C "set transmission subsystem targets" are independent tasks and thus can be executed in parallel. In this example we can see that many tasks are interdependent and thus coupled. Coupled tasks are indicated by marks above the diagonal line. These marks above the diagonal represent design iterations or feedback loops. In fact, all marks below the diagonal are forward feeding tasks and all marks above the diagonal are feedback tasks. Iteration is inherent in the design of any complex engineered system. As stated, the main advantage of the DSM over other forms of product development process modeling is that these inherent feedback loops are easily represented and identified. The magnitude of coupling, and thus complexity of the process, can be seen by the size of a "coupled block." A large coupled block is shown in figure 5.3 between tasks D, E, F, H and I. Therefore we can see from this DSM that the process of designing the transmission to driveshaft interface is highly coupled and iterative. Later we will describe how to break these coupled task blocks down into smaller more manageable couplings when we address how to use the DSM to improve the product development process.

		Set Vehicle level Targets	Set Driveshaft Subsystem targets	Set Transmission Subsystem targets	Driveshaft Slip yoke design	transmission output shaft design	CAE for Driveshaft slip yoke design	CAE for trans output shaft design	driveshaft & trans key life bench test	Run Driveline NVH vehicle Testing
		A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**
Set Vehicle level Targets	**A**	A								
Set Driveshaft Subsystem targets	**B**	X	B							
Set Transmission Subsystem targets	**C**	X		C						
Create Driveshaft Slip yoke design	**D**		X		D	X	X		X	X
Create transmission output shaft design	**E**			X	X	E		X	X	X
Run CAE Analysis of Driveshaft slip yoke design	**F**				X		F		X	
CAE Analysis of trans output shaft design	**G**					X		G	X	
driveshaft & trans Key life bench test	**H**	X	X	X	X	X			H	
Run Driveline NVH vehicle Testing	**I**	X	X	X	X	X				I

Figure 5.3: Example Activity-Based DSM: The Driveshaft to Transmission Interface Design

It is important to note that there are two categories of iteration; planned iteration and unplanned iteration. Planned iterations are those that are required and converge to a high quality design solution. These planned iterations are essential to the design process. The goal is not to eliminate these iterations. Rather, the goal is to shorten the time to complete a single iteration and to decrease the total number of iterations required. Better CAE and CAD are examples of ways to shorten the tasks in an iteration, while better upfront planning and target-setting are examples of ways to reduce the number of iterations. Unplanned iterations are undesirable. These iterations are due to design mistakes, failed validation tests and late changes in program assumptions or targets.

Looking at figure 5.3, we can learn how to read the DSM. Task "I," "Run Driveline NVH vehicle testing" requires input information from tasks "A, B, C, D, and E." Task "C", "Complete Transmission Output Shaft Design" feeds information into tasks "D, G, H and I."

5.4 EXPANDED DSM MODEL FOR PROCESS SIMULATION

The binary DSM has been expanded upon by several academics into a numerical stochastic DSM that holds much more information about the process such as deterministic or stochastic task times, task probability of rework, learning curve effect on time to rework a task, and task sensitivity to changes in input tasks. This additional information brings more insight to the process modeled by the DSM and allows users of the DSM to identify process tasks and iterative loops that cause the most timing and quality issues for the project team. For example, teams evaluating the DSM can ask: Where do the highest probabilities of rework occur? What are the longest tasks? Additionally, this process information can be fed into various existing DSM modeling programs that have been created to help predict total time to complete an iterative engineering design process. Once a valid DSM model of the current process is obtained, this model can be used to test process improvements and approximate the percent reduction in process time that can be achieved [Zambito 2000, Noor 2007].

For this thesis, data is collected on task duration, task information variability, rework time with learning curve, task overlap, task sensitivity, and rework probability. Each of these task quantifiers are defined below and some are shown in the DSM structure in figure 5.4.

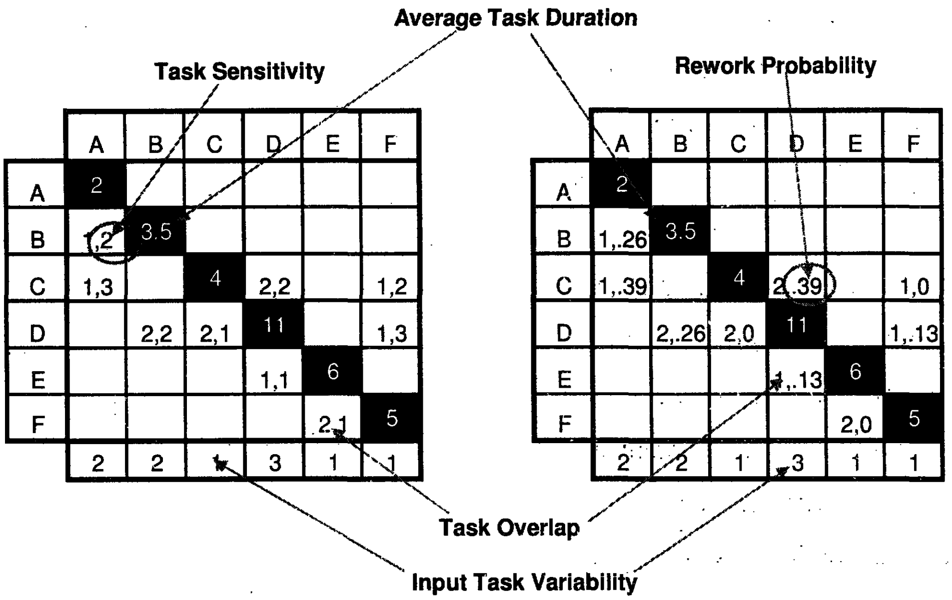


Figure 5.4: Expanded DSM with Task Quantifier Information

Task Duration: Units used are days. Shown for each task in diagonal area of DSM. Interview data was used to create a triangle distribution with average, most optimistic/shortest expected actual work time to complete, and pessimistic/longest expected time to complete. When collecting information on task duration, it is critical to distinguish between actual task time and calendar time. Each yields interesting, but very different, information about the process. In this thesis I collected actual task time.

Task (as an input) Information Variability (IV): No units Shown across bottom of DSM and used to calculate Task Volatility (TV) and probability of rework. This is defined as how likely the information, produced by this task, is to change due to 1) competitive business strategy decisions or 2) original information generated by this task was in error or didn't meet criteria when tested. The IV is quantified by the rating scale shown in Table 5.2.

IV Rating	Description	Estimated Liklihood of change
1	stable	25% or less
2	unknown	between 25% & 75%
3	unstable	75% or greater

Table 5.2: Task Information Variability Rating Scale [Source: Zambito, 2000]

Task Rework Learning Curve: Units %. Data stored on task data collection sheet and is not shown on DSM. When a task is repeated due to required iterations, it does not necessarily take the same length of time that it originally did. Often, there is a learning curve and task completion time is shorter for subsequent task iterations. Not only is there learning, but in fact it may not be necessary to start the task over from the beginning. Therefore, this task quantifier collects data on the average time needed for second and third time performing a task, expressed as a percent of the first and second time respectively.

Task Overlap: Units None. Overlap strategies are discussed later in this chapter. Task overlap is shown on DSM at the intersection of two dependent tasks as first of two digits. This parameter tells us if two dependent tasks can be overlapped or if they must be performed sequentially. This parameter takes the value of 1 or 2. A value of "1" means that the input task must be 100% complete before feeding information to a dependent task and thus the tasks cannot be overlapped. Tasks must occur sequentially if the upstream task information is slow to evolve or the downstream task information is very sensitive to changes in the upstream task. A value of "2" means that an input task only needs to be partially complete before feeding information to a specific task. It is possible to indicate the percent overlap of tasks. Some DSM simulation models take into account the percent of overlap between two tasks [Cho, 2001]. Tasks can be overlapped if upstream task information evolves quickly, can be disaggregated, and fed to a downstream task prior to the entire task being completed. Or, tasks can be overlapped when the downstream task is not sensitive to change in the upstream task information and thus can receive early information assumptions from the upstream task. These overlap strategies are discussed in more detail later in this chapter.

Task Sensitivity (TS): No units. Shown on DSM at the intersection of two dependent tasks as the second of two digits. This task quantifier indicates how sensitive a dependent task is to changes of information from input tasks. If a specific input task changes, how likely is it that a specific dependent task must be reworked? This likelihood is indicated by a Task Sensitivity Rating. The Task Sensitivity Rating Scale is shown in table 5.3.

Rating	description	Dependent task is...
1	Low	Insensitive to most information changes
2	Medium	Sensitive to MAJOR info changes ONLY
3	High	Sensitive to most information changes

Table 5.3: Task Sensitivity Rating [Source: Zambito, 2000]

Probability of Rework: No Units. Shown at intersection of task dependency as the second of two digits. This thesis derives the Probability of Rework from a combination of the *Task (as an input) Information Variability* and *Task Sensitivity*. How likely a specific task is to be repeated is dependent on both how likely the information in an input task is to be changed or be in error and how sensitive that task is to changes in that input task. Therefore, each dependency within the DSM will have its own unique *Probability of Rework*. Dependency marks such as "1" or "X" are replaced with a *Probability of Rework*. The value assigned to a dependency's Probability of Rework is derived from Task Volatility which is the product of *Task Information Variability (IV)* and *Task Sensitivity (TS)*. [Zambito, 2000]

Task Volatility = TV = IV x TS

Then, TV values are assigned a *Probability of rework*. This is shown in table 5.4. The probability of rework vs. TV value curve used to derive probability of rework for each dependency was derived from previous work [Zambito, 2000]. In this previous work, the probabilities were derived from calibrating a DSM model until the probability of rework values entered into the model yielded a model output time consistent with real life vehicle hood development at a NA OEM. This thesis makes the initial assumption that the rework probability vs. TV curve shape from Zambito's work will accurately represent the rework probability vs. TV curve for 2nd order NVH vehicle development. This assumption is made because these two processes occur within the same enterprise type and with the same resource type. The probability values comprising this curve were then calibrated such that the total process lead time for 2nd order NVH development coincided more closely with actual development time.

TV Value	Probability of Rework
1	0
2	0
3	0.07
4	0.13
6	0.20
9	0.26

Table 5.4: Task Volatility vs. Probability of Rework

Data to populate the expanded DSM was collected from interviews with many engineers across several truck programs. Then, this information was used to populate a Microsoft Excel based simulation program available on the website dsmweb.org.

The purpose of using a DSM PD process simulation model is to quantify the predicted percent improvement in process timing of various process improvement proposals. Due to the scope, this thesis will be less concerned with calibrating the simulation absolute value output with real life process time and more concerned with the percent improvement seen between the "As-Is" and "to-be" process simulations. The simulation will help assess which process improvements are most effective and approximately how much improvement can be expected.

In order for the product development process model created in this thesis to have useful predictive value, it must meet several criteria. First, it must address important managerial issues. This model does indeed address important managerial issues as it attempts to improve a process that in the past has caused program delays and led to designs that cause field warranty issues. Second, the decision making is based on information that is available and accurate. The process model of 2nd order NVH attribute management at NA OEM was created from interviews with engineers involved in the processes of study. Input from engineers was averaged across several input values from various engineers to ensure the data was more accurate. Third, the assumptions and simplifications of the model are reasonable. The assumptions of the model can be considered reasonable. This assertion will be supported in chapter 6. However, these assumptions are reasonable for the type and use of model output. Finally, the model must have face validity and can be correlated with previous projects. In chapter 6, the "As-Is" model output appears to be reasonable as compared to the real process. Although the "As-Is" output is longer than the product development cycle time allows, this is due to the fact that, in reality, not all of the tasks are completed as thoroughly as they should be as engineers, short on time and resources, triage their work [Smith & Morrow, 1999, Noor, 2007].

The output of the Activity Based Product Development DSM simulation models, used to evaluate and control projects, is not the only useful part of the model. In addition, the process of creating and understanding the model can help a cross functional product development team create common goals and a better understanding of tasks. One should always consider the indirect benefits of a model [Smith & Morrow, 1999].

5.5 Creating the Expanded Activity-Based DSM

The DSM creation process can be thought of in a three phase approach. These three phases are 1) Knowledge Gathering, 2) Analyzing and Optimizing, and 3) Documenting and Communicating [Zambito, 2000]. In this thesis, the 2nd order NVH DSM creation process was found to be just as insightful as the DSM analysis and simulation process.

Once a significant management problem is identified, a team assembled, and the Activity-Based DSM chosen as the method of analysis, the first challenge is to identify the DSM scope. The team must identify the specific system, subsystems or design problem being studied. For example, the scope of the DSM created for this thesis includes all of the tasks that define the subsystem parameters which contribute directly to 2nd order NVH. The team must decide which systems, organizations and processes are "in scope", one level out, and "out of scope." In general, it is best practice to include those systems, organizations and processes that are directly in scope and those that are one level out. [Zambito, 2000]

Next the team must create the comprehensive list of tasks that comprise the process. Here, the team must decide what level of detail is best suited for DSM. They must decide how many tasks the process should be broken up into. Does the team want 10 high level tasks or 50 low level tasks? Which tasks can be combined into a single task to simplify the process model? Generally, a process flow diagram can help the team better understand the process and approximate tasks dependencies and the level of detail for each task. Task detail should be minimized unless it hides essential information. As a general rule, if multiple tasks have the same upstream and downstream relationships, they can be combined into a single, higher-level task [Zambito, 2000]. Also, the process flow diagram can be a good basic communication method when speaking with engineers about the process tasks they own and how they might tie into the entire process. However, tasks should be separated if they involve different organizations, have different timings or different levels of uncertainty in the various task criteria recorded.

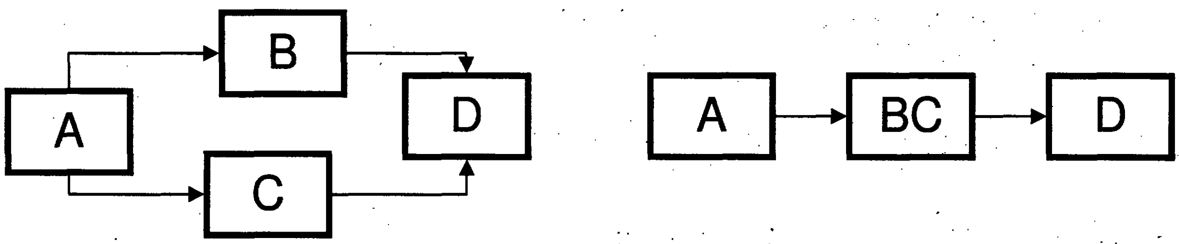


Figure 5.5: Combining Tasks to Achieve Optimal Level of Detail

The quality and timeliness of the DSM building process can be improved by a task deliverables list consensus meeting. Most often, perceived task input and output deliverables do not match between various engineering teams. If all teams can meet to establish a single agreed upon list of deliverables, the process of populating the DSM with dependency marks will be greatly expedited. In addition, significant gaps in mutual understanding may be discovered and filling those gaps will result in an improved process. The task list will not need to be modified as various teams are interviewed. Also, a single agreed upon task list can facilitate electronic interviews [Guivarch & Whitney, 2004].

Once a task list is created, the team must collect data for each task. A chart should be created that shows each task, its associated deliverable and who is responsible for completing the task and owning the deliverable. This chart can also be used to collect data on task duration, task sensitivity and task learning curve.

Finally, the DSM can be populated with task information. Task dependencies can be recorded in the DSM by completing interviews with the responsible engineers. Generally, it is best to ask engineers "What information do you need to complete this task?" and then "Who needs the information you create?" By asking engineers these two essential questions, the team can populate the DSM and verify that their task list is comprehensive and at a correct level of detail. As stated, this process can be improved by starting the process with a single list of task deliverables agreed upon by all of the teams involved in the DSM creation process. Once the DSM is populated with dependency marks, planned iterations can be highlighted with a solid box. Unplanned iterations can be outlined by dashed boxes.

After the "As-Is" process DSM is completed, the team can begin to analyze the information in the DSM. Important insights gathered while creating the "As-Is" DSM can be documented. The next section will describe how the "As-Is" DSM can be analyzed and discuss some strategies for optimizing the process. However, it is important to note that the mere process of creating the "As-Is" DSM can lead to many useful process insights.

5.6 DSM Analysis & Methods for Improving Process Time & Product Quality

Once the "As-Is" DSM is created, it serves as a powerful tool for understanding the current process and for devising process improvements. The team will have a clear representation of how information flows through the process. They will understand information inputs and outputs for each task. Also, they will be able to easily identify iteration in the design process. The team can then focus on eliminating unintended iteration. For intended planned iterations, the team can attempt to execute faster iterations or to conduct fewer iterations. There are many DSM algorithms and analysis techniques to improve the process by moving as many dependency marks to the lower half of the DSM and eliminating as many upper half dependency marks as possible. The main algorithms and techniques are discussed in this section. In addition, a careful DSM analysis can identify process mistakes such as redundant tasks, premature decisions and tasks that are starting too late in the process. Also, process experts can study the DSM and identify restructuring opportunities that achieve a reduction in task time and rework probability

5.6.1 Partitioning the DSM

Sometimes, the tasks in the DSM can be resequenced such that an improved process structure is revealed. Deliberately changing the task sequence in the DSM is referred to as partitioning or sequencing. The sequencing operation of the DSM attempts to minimize the number of dependency marks that occur above the diagonal. The goal is to only have iterative dependency marks, marks above the diagonal, that are due to true task coupling and cannot be reduced by partitioning alone [Steward, 1981]. Some feedback loops in an initial DSM are only due to process sequence and if tasks are simply rearranged, then these feedback loops disappear from the process. Many DSM partitioning algorithms exist to assist DSM users. Figure 5.6 show an example of a DSM before and after partitioning.

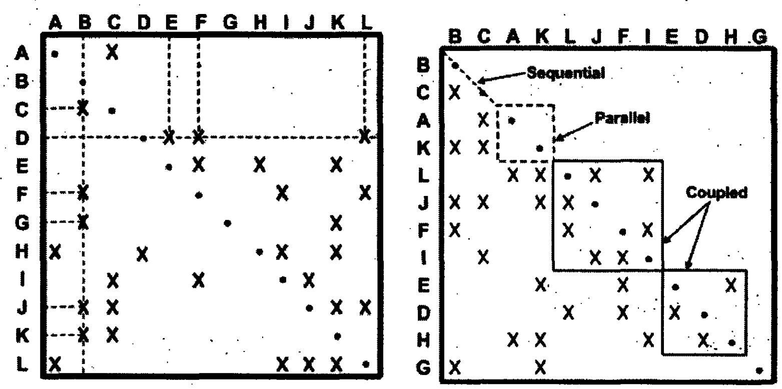


Figure 5.6: DSM Before vs. After Partitioning [Source: DeWeck, 2006]

5.6.2 Tearing the DSM

Often after attempting to partition the DSM, large coupled blocks still exist. These coupled design problems involve many iterations and engineering teams are often faced with "chicken or the egg" type decisions. One way to break large coupled task blocks into more manageable iterative task sequences is to make a design assumption or to standardize the design. When making an assumption on the information produced by a task, it is critical to ensure that the iterative design loop has low sensitivity to this assumption in case the assumption is not entirely correct. When these assumptions are made as part of a process change, it is reflected on the DSM as "tearing." A single large coupled block is torn into multiple smaller coupled blocks. Figure 5.7 shows a DSM before and after "tearing." The tear mark is shown where assumptions were made about missing information. Algorithms have been created to optimally tear a DSM, so that marks above the diagonal are minimized.

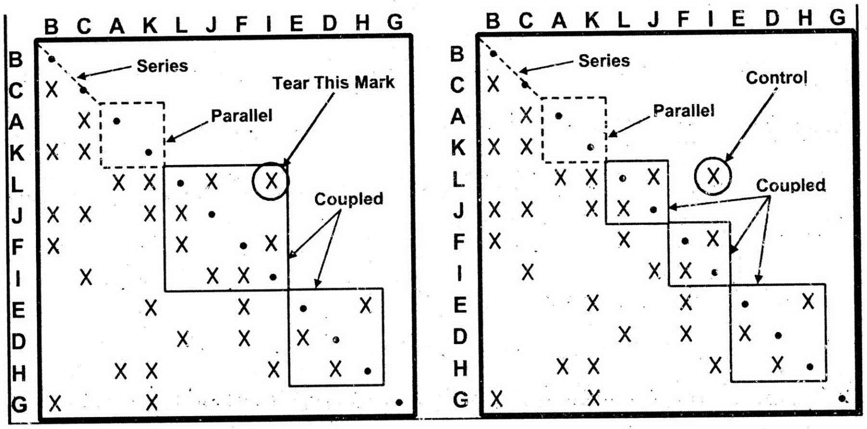


Figure 5.7: Tearing a DSM [Source: DeWeck, 2006]

5.6.3 Overlapping of tasks

Overlapping product development tasks can help firms develop products faster. However, the overlapping of product development tasks must be carefully managed. Without careful management, overlapping tasks may actually lead to increased development times and resource requirements. Therefore, one cannot simply say "overlap tasks as much as possible." Instead one needs to use criteria that have been developed to determine if and when tasks should be overlapped. [Krishnan et al., 1997].

Often two tasks cannot be executed in parallel because a dependency exists. However, these two tasks may be overlapped to reduce total execution time following the overlap criteria developed by Krishnan, Eppinger and Whitney. "Overlapping" means the downstream activity begins earlier by using preliminary information. When two tasks are over lapped, the upstream task information "X" needs to be disaggregated into parts "X1 and X2." See figure 5.8 below.

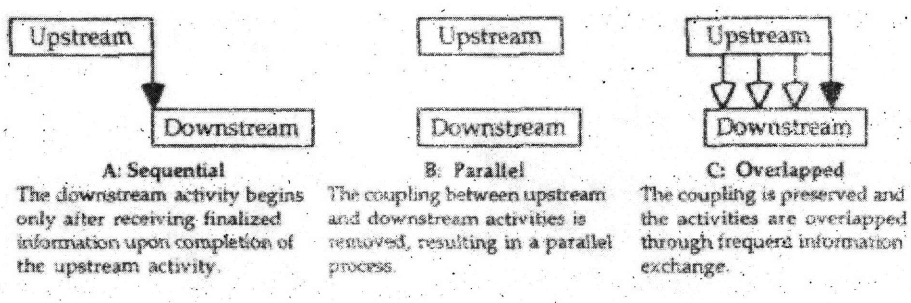


Figure 5.8: Sequential, Parallel, and Overlapped Tasks [Source: Krishnan et al., 1997]

When devising a task overlap strategy, management can first consider the extreme values of Upstream Information Evolution and Down Stream Information Sensitivity and then use the framework presented in figure 5.9 to devise a product development activity overlap strategy.

Upstream information evolution: The rate of refinement of the upstream generated information from its preliminary form to a final value.

Downstream information sensitivity: The relationship between the duration of downstream iteration and the magnitude of the change in the upstream information value. If the downstream task is highly sensitive, then larger changes in the value of the exchanged information require longer iterations to process those changes. [Krishnan et al., 1997]

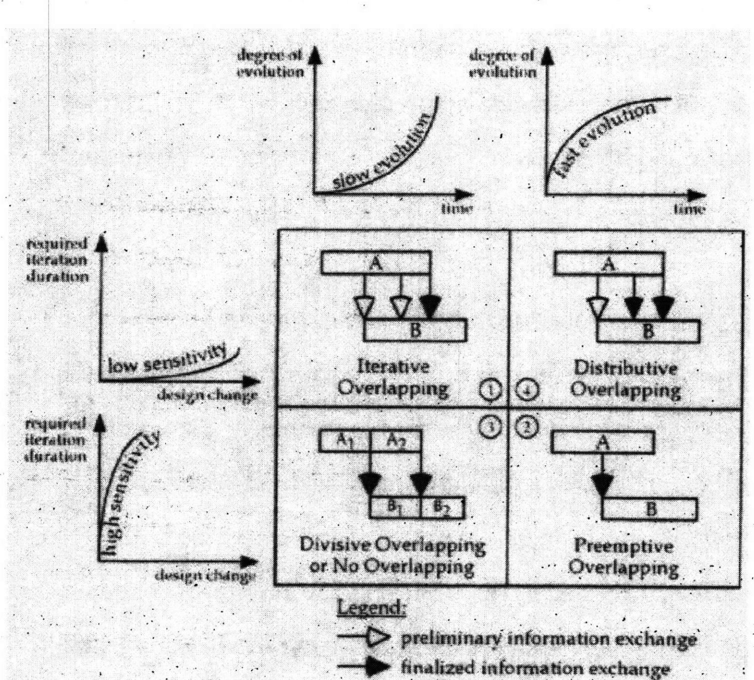


Figure 5.9: Framework for Managing Task Overlap [Source: Krishnan et al., 1997]

Figure 5.9 shows four prescriptive strategies that product development management can use for overlapping formerly sequential tasks. The strategies depend on envisioning the extremes of task evolution and sensitivity. Questions that need to be asked are: How quickly does the information in the upstream task converge to a final value? Does the information evolve and converge quickly and remain relatively the same value throughout the task duration? Or, does the upstream information evolve slowly and only converge to a well defined value at the very end of the task? The next questions that should be asked are: How sensitive is the downstream task to changes in the information that is fed from the upstream task? If the upstream task information changes, does this cause a high magnitude of rework for the downstream task? Or, do changes in the upstream task information cause little rework for the downstream task? Based on the answer to these questions, four overlapping strategies are described below:

Iterative Overlapping: Slow upstream evolution and low downstream sensitivity. The activities are overlapped by beginning downstream activities with preliminary information, and incorporating design changes in subsequent downstream iterations.

Preemptive Overlapping: Fast upstream evolution and high downstream sensitivity. The information in the upstream task converges quickly. Thus, the upstream information can be frozen "preemptively" and fed to the downstream task. Here the process becomes shorter because the downstream task would start earlier, but with finalized upstream information.

Distributed Overlapping: Fast upstream evolution & low downstream sensitivity. This is the best scenario for overlapping. Downstream tasks with low sensitivity can start with advanced upstream information and some of this upstream information can even be preemptively finalized due to the fast evolution.

Divisive Overlapping or No Overlapping: Slow upstream evolution and high downstream sensitivity. It is not desirable to start downstream tasks with preliminary upstream information because the upstream information is evolving slowly and will most likely change. In addition, the changes in upstream information will create a large amount of rework for the sensitive downstream task. Thus, no overlapping is recommended. However, if the upstream task information can be disaggregated into information that is finalized quickly and information that is finalized later, then pieces of information that are finalized early can be fed to the downstream task.

5.6.4 Do-it-right-the-first-time (DRFT)

In many cases, partitioning of a DSM by re-sequencing of process tasks is not possible. Or, partitioning only yields mild process improvements. In order to achieve more drastic process improvements, the tasks within the DSM that cause large iterative loops need to be studied by process experts. Entirely new tasks may be needed. Current tasks may need to be redefined. Current tasks may need to be divided into multiple tasks and some tasks may be deleted. All of these actions can be used to "re-engineer" the process to yield large improvements in process time and organizational resources.

When applying a DRFT approach to DSM restructuring to achieve an improved process, one should [Yassine et al., 2000]:

1. Create the base ("As-Is") DSM so that iterative loops can be identified and understood.
2. Apply a partitioning algorithm to see if any improvements can be made to the base DSM.
3. Identify "design-and test" cycles in iterative blocks.
4. For each "design-and test" cycle, decide if a new DRFT task can be inserted at the beginning of the block. The DRFT task might consist of the application of an expert system that improves the process by enabling it to start with more accurate information.
5. Create a DSM of the new process, measure improvements and compared to baseline DSM.

5.6.5 Identifying Bottlenecks and Critical Inputs

Figure 5.10 shows task "B" as an example of a *Critical Input*. This input is critical because the completion of many tasks, immediately downstream in the process, rely on task "B" to be completed and output information to be made available. Also, if task "B" is completed incorrectly or the information is wrong, these downstream tasks mostly likely need to be repeated. The criticality of the timeliness and correctness of task "B" information means this task's deliverables should be standardized if at all possible. In addition, this task should be noted as critical and not deliberately changed because so many down stream tasks depend on this task. The most critical tasks can be identified by creating a Pareto chart depicting the number of dependant tasks associated with each task. The number of marks per column (input task) can be counted to create the Pareto chart. [Noor, 2007]

Figure 5.10 also shows task "E" as an example of an information bottleneck. Here a single task must wait for input information from many other tasks. Thus many upstream tasks must be completed before this downstream task can begin [Noor, 2007]. The largest bottlenecks can be identified by counting the number of marks per row (output task) and creating a Pareto chart with this data.

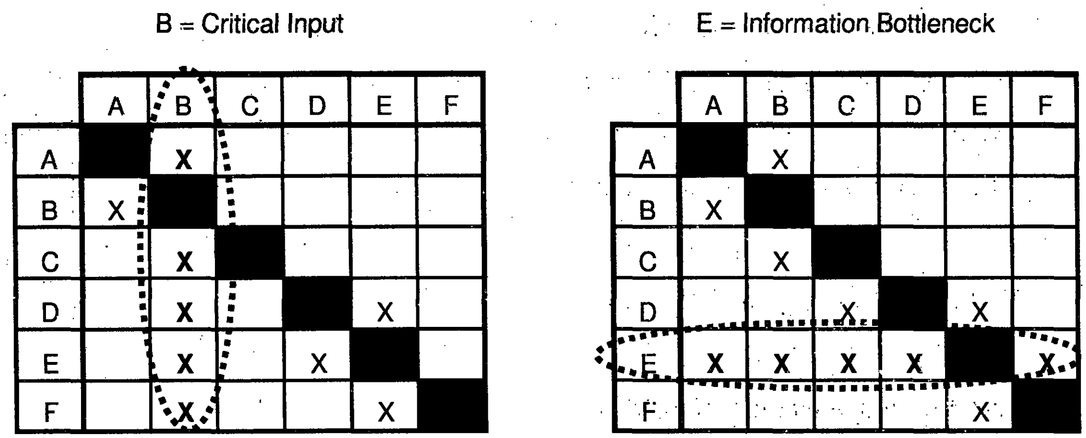


Figure 5.10: DSM Example of a Critical Input and an Information Bottleneck

5.7 Conclusion

All of the DSM analysis strategies discussed above will be used to study the current process of delivering a vehicle design that meets second order NVH attribute targets and to propose improvements to this process that will result in shorter process cycle time and improved vehicle system quality. This analysis is presented in the next chapter.

6. APPLYING THE DSM METHOD TO ATTRIBUTE MANAGEMENT PROCESS IMPROVEMENT

6.1 Introduction

As discussed in the previous chapter, we have chosen to examine the current process of designing and integrating the appropriate subsystems to deliver a vehicle system that meets 2nd order NVH attribute targets. Essentially, we are asking "What are the current processes of 2nd order NVH vehicle level attribute management and how can these processes be improved?" We have chosen to use the systems engineering process modeling method, the Design Structure Matrix (DSM) as the means to understand the current process and test improvements. The DSM reveals significant opportunity for process restructuring, redesign and improvement that promotes improved cross organizational design efforts and will enable a higher quality design to be completed in less time than before.

6.2 Creation of the "As-Is" DSM Process Model

6.2.1 Scope and Granularity

The scope of this DSM includes all tasks executed by product development teams that contribute to 2nd order NVH. This includes all teams that engineer subsystems that have design parameters that control the NVH source, magnitude, transfer function, and response. It also includes the following additional engineering support teams: basic design, the team that defines vehicle architecture and complexity early in the program; vehicle engineering which is responsible for understanding and controlling vehicle transfer functions, for cascading vehicle level targets to subsystems; CAE, responsible for early analytical testing; and driveline systems, responsible for the integration of the driveline subsystems and vehicle prototype testing. The granularity is such that subsystem level design tasks are listed. However, individual subsystem level design tasks are grouped whenever possible. For example, we state "define rear axle design parameters" as opposed to listing each rear axle design parameter as a separate task. These design parameters have the same inputs and outputs and thus can be grouped as a single task. The goal was to model 2nd order NVH attribute design in a task-based DSM that would not overwhelm the driveline organization, which is using this systems engineering method for the first time. Thus, it was decided to pick a granularity that produced approximately 40 tasks plus or minus 10 tasks.

6.2.2 Process Flow Chart

The flow of information between teams was initially understood by the creation of a process flow chart shown in figure 6.1. This initial process flow chart served as a good communication tool while trying to understand how information flows both within and between PD teams. The flow chart shown in figure 6.1 represents the teams initial view of the "as-is" process of managing 2nd order NVH. The arrows represent the direction of information flow in the process. The black arrows represent feed forward information flow and the red arrows represent information feedback loops. It is interesting to note that many more rework loops were discovered after the "as-is" DSM was created.

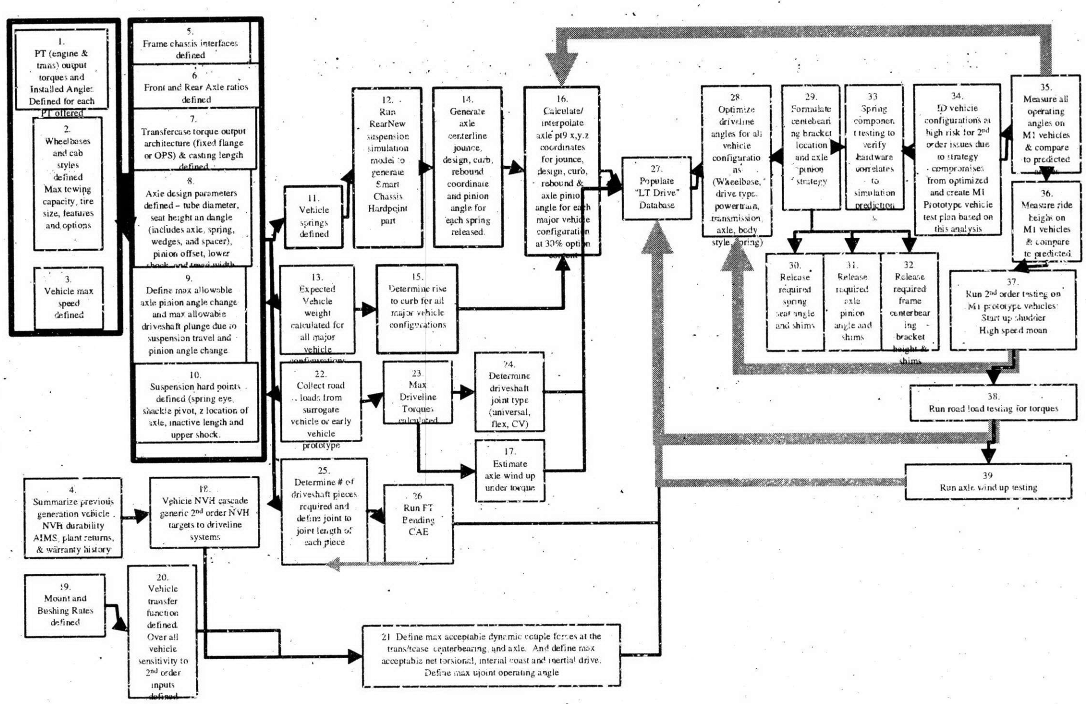


Figure 6.1: 2nd order NVH Attribute Management Process Flow Diagram

6.2.3 Task List

The final task list for the As-Is DSM was created by interviews with several engineers working on one of two major truck programs from each individual team. In hind sight, the process for creating the final task list could have been drastically improved if an initial "deliverables agreement" meeting had been held with all teams at once. If this meeting had been held, all teams could agree at one time to the information they consume and produce and the deliverable format, name and storage location for this information [Guivarch & Whitney, 2004]. In this thesis case, the task list was modified all throughout the DSM creation process which made updates and version tracking difficult. The final As-Is task list, consisting of 35 total tasks, includes deliverable name, responsible engineering team, and data on task time, rework learning curve and task variability.

6.3 As-Is DSM

6.3.1 Planned Iterative Blocks

After following the "As-Is" DSM creation process discussed in chapter five, the DSM, shown in figure 6.2, was created. There are 35 total tasks that comprise the 2nd order NVH attribute management process at the desired level of granularity. The original as-is process flow chart shown in figure 6.1 contains 39 tasks. The "as-is" DSM has only 35 tasks because the team found that some of the tasks could be combined. The unintended iterations, due to failed vehicle level prototype verification tests, are shown as above diagonal marks in light grey font in columns 29 through 35. Excluding the unintended iterative dependency marks, two main coupled iterative blocks of intended iterations become apparent. The iterations in these blocks are intended as part of the normal vehicle design process. However, these two blocks are highly coupled to the extent that it is difficult to discern between the two blocks and thus should be considered one iterative block in the "As-Is" DSM analysis. These two highly coupled blocks represent the two major highly coupled iterative design processes involved in delivering a vehicle design that meets 2nd order NVH targets. The first iterative design process, "Block I" in figure 6.2, represents the iterative design process of creating the transfer case, transmission, axle and driveshaft design and setting the vehicle max speed such that the driveline meets powertrain bending requirements. Powertrain bending is a result of the driveline system reaching resonance at a given vehicle speed. The driveline powertrain bending requirement is verified with computer aided engineering tools early in the design process and is validated with vehicle testing later in the design process.

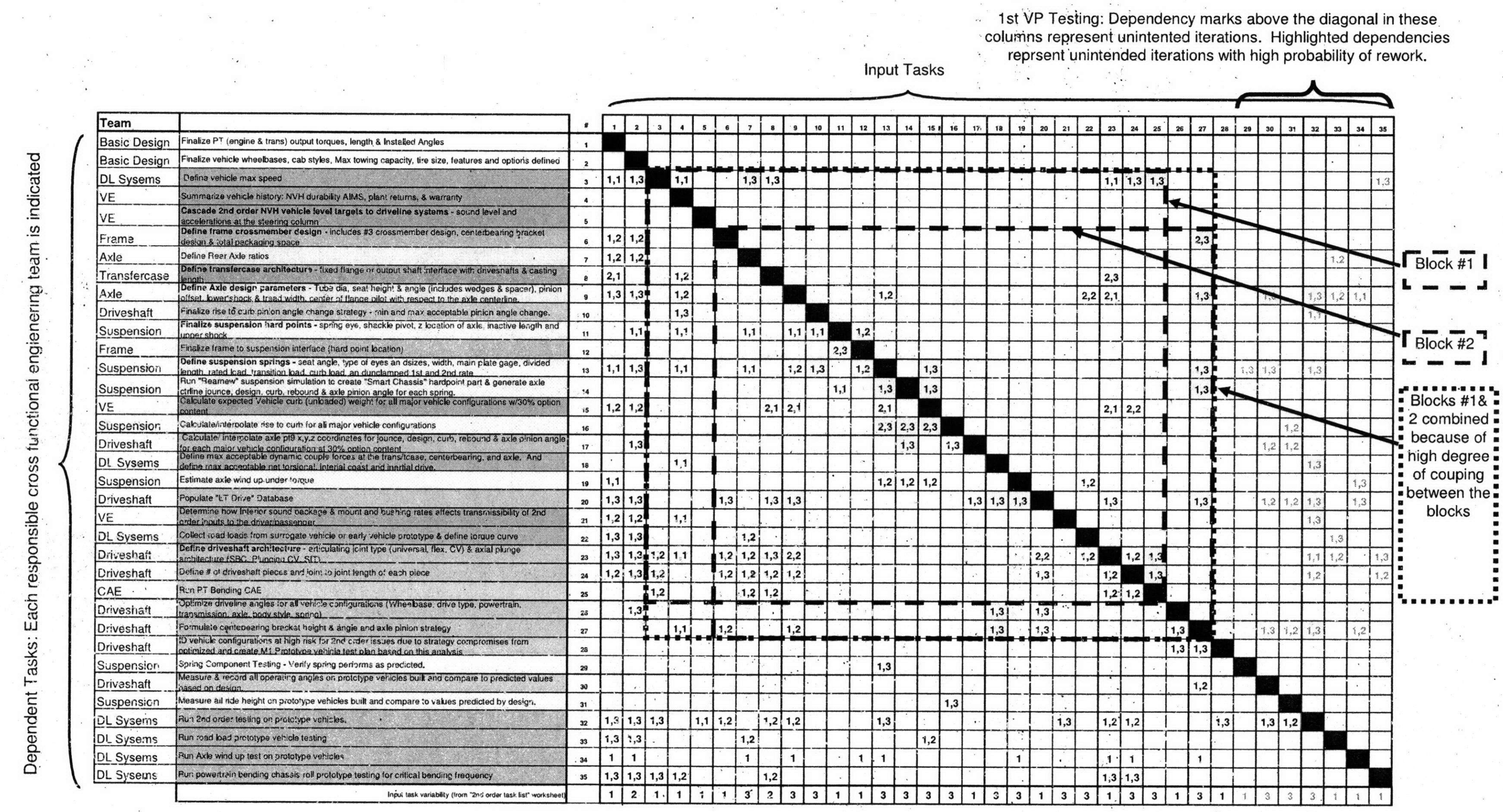
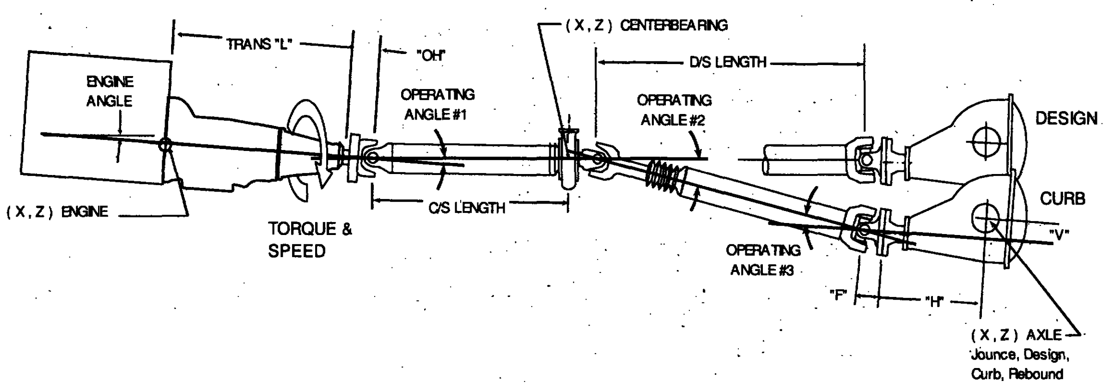


Figure 6.2: DSM Representing Current "As-Is" 2nd order NVH Attribute Management Process

The second iterative block, shown as "Block 2" in figure 6.3, represents the iterative design process between the driveshaft, rear axle, suspension, frame, powertrain, transmission and transfer case teams to set truck driveline angles for all vehicle weight conditions between fully loaded and empty and all driveline torque conditions. The driveline angles control the magnitude of the source of 2nd order NVH. As discussed in chapter four, the source of 2nd order NVH is the driveshaft single cardan universal joint. The magnitude of the forcing function is dependent on the driveline angles, torque and speed. Driveline angles are set up by the powertrain installed angle, driveshaft mating position with the transmission or transfer case, driveshaft center bearing mounting position on the frame cross member, the rear axle position and angle as determined by the suspension in various vehicle weight conditions. The driveline angle and associated subsystems that contribute to driveline angles are shown in figure 6.3. The response at the driver and passenger location is dependent on total vehicle sensitivity and the vehicle "sound package" which is partially controlled by upper body design.



OH = U-joint Overhang
Trans "L" = Transmission length
C/S = Couple Shaft of the two piece driveshaft system
D/S = Driveshaft
X,Y,Z = Vehicle Coordinates
"F" = U-joint center to flange end length
"H" = Rear axle pinion length from axle centerline to pinion flange end
"V" = high from center of pinion flange to centerline of axle

Figure 6.3: Driveline Angles and Determined by Powertrain, Frame, Suspension and Rear Axle

6.3.2 Unplanned Iterations

Iterations that occur because of failed vehicle level prototype testing have been marked as unplanned iterations. The corresponding dependencies have been marked as unplanned iterations because, according to NA OEM NEW PDS, the underbody design must be at design intent by the first vehicle prototype (1^{st} VP) build. NEW PDS has two main prototype builds for most programs of varying complexity. However, for programs with almost entirely carryover content or with almost entirely new content there are one or three total prototype builds respectively.

The majority of new programs have two prototype phases; we will refer to these prototype phases as 1^{st} VP and 2^{nd} VP. The 1^{st} VP build requires that the underbody and powertrain be at design intent while the upper body design is still not finalized. The offset in design progression at 1^{st} VP between the under and upper body is intentionally created to allow resolution of powertrain and underbody issues without forcing changes to the upper body and creating rework [NA OEM, 2007]. However, this offset does lead to some potential issues for underbody design parameters that control vehicle functionality and attributes and are affected by upper body characteristics and design parameters. This issue is resolved by calling these types of design parameters, "tunable" which are further developed during the upper body design finalization phases. The 2^{nd} VP build requires that all systems are at production intent. For most vehicle programs, all of the underbody subsystems contributing to 2^{nd} order NVH must be designed to production intent and integrated at the 1^{st} VP build and verified during 1^{st} VP testing. Changes to underbody design parameters are highly undesirable after 1^{st} VP testing. Underbody subsystems contributing to 2^{nd} order NVH are the frame, suspension, transmission/transfer case, rear axle, and driveshaft. 1^{st} VP testing must verify that the vehicle system meets 2^{nd} order NVH attribute targets.

Marking 1^{st} VP dependencies that occur above the diagonal as "unplanned" allows the DSM to be partitioned based on planned iterations only. This yields smaller iterative blocks of planned iterations only. However, unplanned iterations cannot be ignored as part of our DSM process analysis because they can cause major delays to a program. In this case study, the vehicle system very often fails to meet 2^{nd} order NVH attributes targets at 1^{st} VP and the team has to execute unplanned iterations between the 1st VP and 2nd VP prototype testing. The high probability of unplanned iterations is due to many process and organizational issues that will be addressed in this thesis. This DSM highlights unplanned iterations that have a high probability of rework as areas that need problem resolution by the process restructuring team. This thesis will discuss the root cause and potential process and organizational corrective actions that could be implemented to prevent these unplanned iterations from occurring after 1st VP testing.

6.3.3 Cross Organizational Aspect of As-Is DSM

Each task in figure 6.3 is color coded to represent the engineering organization responsible for the completion of that task and the creation and storage of the corresponding deliverable. This color coding shows the highly cross-functional nature of the management of this attribute. The PD engineering teams that control design parameters, decisions and testing affecting 2nd order NVH are the "Basic Design," Powertrain, Vehicle Engineering, CAE, Suspension, Frame, Transfer case or Transmission engineering, Rear Axle engineering, Driveline Systems, and Driveshaft teams. The cross functional nature of 2nd order NVH attribute management creates potential issues with deliverable hand-offs, communication, timing, complexity, attribute and parameter trade-offs, and understanding parameter design space. As you can see from figure 6.3, the information flow between these organizations is highly coupled. Effective and efficient communication and information sharing amongst all of these organizations is imperative. Also, it is imperative that teams create and access deliverables "on time." "On time" creation of deliverables can occur only when each organization understands the timing constraints of other dependent organizations.

6.3.4 Representing Task Overlap and Probability of Rework

Aside from task dependencies, the As-Is DSM in figure 6.3 reveals information on upstream and down stream task overlapping and the probability of rework. As discussed in chapter 5, this matrix includes information that can be used to aid in a visual analysis of the DSM as well as a simulation of total process time. Each dependency cell shows two values separated by a comma. The first value, taking on the value of 1 or 2, represents, respectively, if the two tasks are executed sequentially or if they are overlapped (refer to discussion in chapter five). The second value in the dependency square can take on the value of 1, 2 or 3 and represents the output task's sensitivity to changes in the input task (refer to discussion in chapter five). At the bottom of the DSM, there is a row labeled "Input Task Information Variability." Each task is assigned a value of 1, 2 or 3 depending on how likely the information produced by that task is to be incorrect or changed (refer to discussion in chapter five). As explained in chapter five, the task dependency sensitivity and the input task information variability are used to determine the probability of rework. The task dependencies with the highest probability of rework (sensitivity =3 and variability =3) are highlighted in yellow in figure 6.3 DSM. The DSM analysis will, in part, focus on these highlighted task dependencies. The DSM analysis team will strategize on how to drastically reduce the probability of rework for highlighted unintended iterations. These highlighted unintended task dependencies are extremely undesirable because they cause significant design rework. Often, these iterative loops will cause program delay even if infinite resources are added to the program [Noor, 2007].

6.3.5 Validation of the "As-Is" DSM

The completed "As-Is" DSM process model was validated by final reviews with all teams involved in the process. Each team agreed that the model represents the process as it is currently executed. A final consensus was drawn on whether or not tasks were overlapped and what the task information variability and dependency sensitivity values should be to correctly represent the current process.

6.4 DSM Simulation

In section 6.5, the As-Is DSM process model will be transformed into the "To-Be" DSM process model. As the DSM is analyzed and process improvements are proposed, a means to test relative changes in process lead time mean, standard deviation and coefficient of variation is required. An MS Excel based computer program, available on the website dsmweb.org, has been selected to enable this comparative analysis. This thesis will be limited to comparing relative change in process lead time mean and standard deviation as well as the change in coefficient of variation (Coefficient of variation = standard deviation/mean).

The thesis is limited to a comparison between the current "As-Is" process lead time and the proposed "to-be" process lead time because the process lead time produced by the simulation could not be directly correlated with the actual process time. The "As-Is" process lead time information will be the established baseline to which improvement proposals will be compared. The percent change in lead time, standard deviation and coefficient of variation will be used as metrics to analyze process improvement proposals. These will be calculated using the following equations:

$$\% \, \delta\mu = [(\mu_{as\text{-}is} - \mu_{to\text{-}be}) / \mu_{as\text{-}is}]*100$$

$$\% \, \delta\sigma = [(\sigma_{as\text{-}is} - \sigma_{to\text{-}be}) / \sigma_{as\text{-}is}]*100$$

$$\% \, \delta C_{\upsilon} = [(C_{\upsilon as\text{-}is} - C_{\upsilon to\text{-}be}) / C_{\upsilon as\text{-}is}]*100$$

The inability to correlate the simulation lead time with the actual process lead time is due to three facts. First, there have not been any formal efforts made at NA OEM to collect accurate process data for the 2^{nd} order NVH attribute management process. All data collected were estimates from engineers responsible for completion of the various tasks in the DSM. Second, only two programs of similar complexity were sampled for data required as inputs into the simulation model and thus this sample is not statically significant. Only two programs were sampled because truck program complexity varies greatly from model to model and from one model year to another. Historical task time and iteration data would be needed from six or more truck programs of similar complexity to make the model data statistically significant. In addition, data would have to be collected from programs of varying complexity to correlate complexity to process lead time. Then, a scaled model of the process that takes into account the complexity differences could be created to more accurately estimate process lead time for all programs. Third, engineers have estimated, separate from other tasks, how much consecutive time, in days; it usually takes them to complete each task. In reality, all engineers work on several different vehicle programs at one time. Thus, their days are split between all programs that they are responsible for. Therefore, the Consecutive Time data collected in this thesis differs greatly from Calendar Time. In this thesis, I derive task time by assuming that a single engineer is dedicated 100% to each task. In reality, a single engineer rarely dedicates 100% of their efforts to a single task from start to completion. Instead, they handle multiple tasks and have to divide their calendar time between many tasks.

6.4.1 Collection of DSM Simulation Input Data

Data to populate the DSM process simulation was collected via interviews with engineers from two major truck programs. Each program has considerable complexity due to a high number of wheelbase, powertrain and drivetrain options.

6.4.2 Task Duration Assumptions

The DSM Microsoft Excel based simulation model computes stochastic process lead time by using task time duration data in the form of a triangle distribution. For each task, three durations are needed, optimistic or shortest expected completion time, average or expected completion time, and pessimistic or longest expected completion time. The following assumptions were made regarding the duration:

1. Duration is specified in days and is equivalent to total time spent on the completion of that task. This duration value is not calendar time; rather it is consecutive time spent completing the task.
2. For each of the two truck programs, optimistic, expected and pessimistic task durations were collected from interviews with one engineer from each responsible team. This is consistent with engineering resources at NA OEM where a single engineer from each team is assigned to a single or multiple programs.
3. Optimistic and pessimistic task times represent 10^{th} and 90^{th} percentiles respectively.

6.4.3 "As-Is" DSM Simulation Results

Once all of the "As-Is" process information was collected and compiled, the simulation was run. The results are summarized in table 6.1. A histogram showing the process lead time results of the 1000 run Monte Carlo simulation is shown in figure 6.4.

"As Is" Process Lead Time Mean (days)	$\mu_{as\text{-}is} = 114$
"As Is" Process Lead Time Standard Deviation (days)	$\sigma_{as\text{-}is} = 19.5$
"As Is" Process Coefficient of variation	$C_{\upsilon as\text{-}is} = 0.17$

Table 6.1: "As-Is" DSM Simulation Results

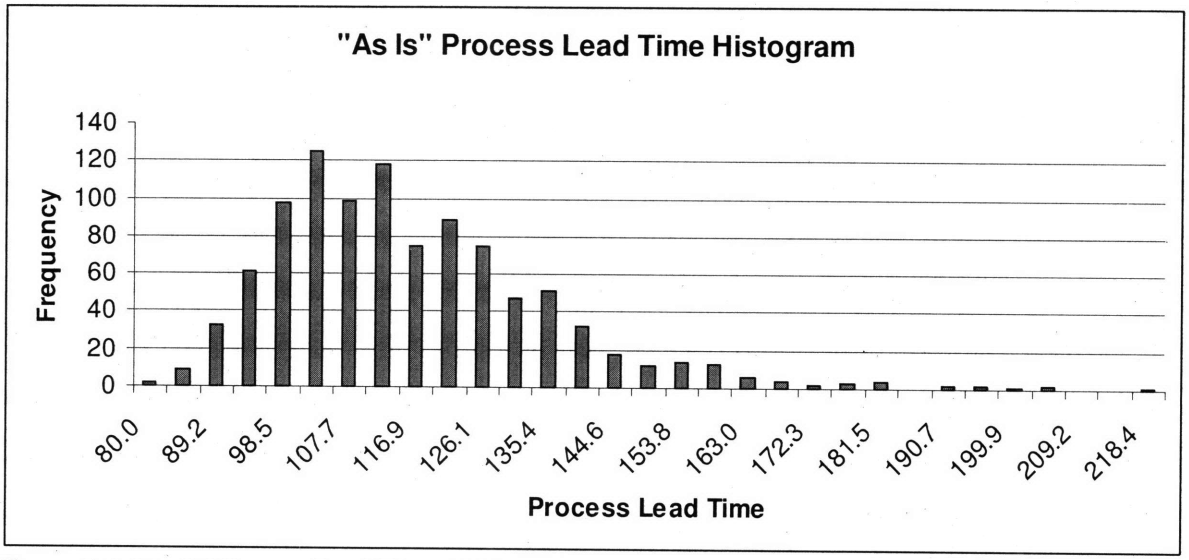


Figure 6.4: "As-Is" Process Lead Time 1000 Run Monte Carlo Analysis Histogram

6.5 Restructuring the "As-Is" DSM: Creation of the "To-Be" DSM

The process of managing 2nd order NVH vehicle level attribute can be improved by studying the "As-Is" DSM and using the DSM analysis methods discussed in chapter five to restructure the process. The process restructuring will be accomplished by a team comprised of process experts from all engineering teams that own design parameters that control 2nd order NVH or support subsystem and vehicle level development and verification. The relative impact of process restructuring proposals on process lead time will be revealed by DSM simulation results for the restructured process. However, it is important to note that the process restructuring goal is to create a process that yields a higher quality system in less development time and within budget. In section 6.5.5, the potential quality impact of the restructured process will be discussed in qualitative terms based on expected system robustness gained from the process improvements. A qualitative discussion on how the restructured process will reduce resources will also be included.

6.5.1 Partitioning the DSM

Prior to any further analysis, the "As-Is" DSM was first partitioned with a partitioning algorithm to see if any tasks could be simply rearranged to yield less feedback dependencies, that is, less marks above the diagonal. The partitioning was done for two "As-Is" DSM scenarios. First, the "As-Is" DSM was partitioned including all 35 tasks. Next, the "As-Is" DSM was partitioned excluding 1st VP validation test tasks. These 1st VP test tasks create the unintended iterations because these vehicle prototype tests are intended to validate the design, not to create additional work. The DSM is partitioned excluding the unintended iterations to see if the intended iterations due to natural design progression could be reduced further if the probability of unintended iterations occurring was low enough to disregard these potential iterations. After partitioning both scenarios, it was found that if the probability of unintended iterations was low enough to disregard these possible dependencies, the iterative block of intended tasks could be reduced from 25 tasks to 18 tasks. This means that the process restructuring team should focus on how to reduce the probability of rework for all unintended vehicle prototype testing iterations. The results are shown in table 6.2. The two scenarios are shown in figure 6.5 and figure 6.6.

Scenario	Intended Iterative Block Diameter
"As-Is" before partitioning	25 tasks
Partitioned: All 35 tasks included	22 tasks
Partitioned: 1st VP Validation tests excluded	18 tasks

Table 6.2: Intended Iterative Block Diameter Comparison

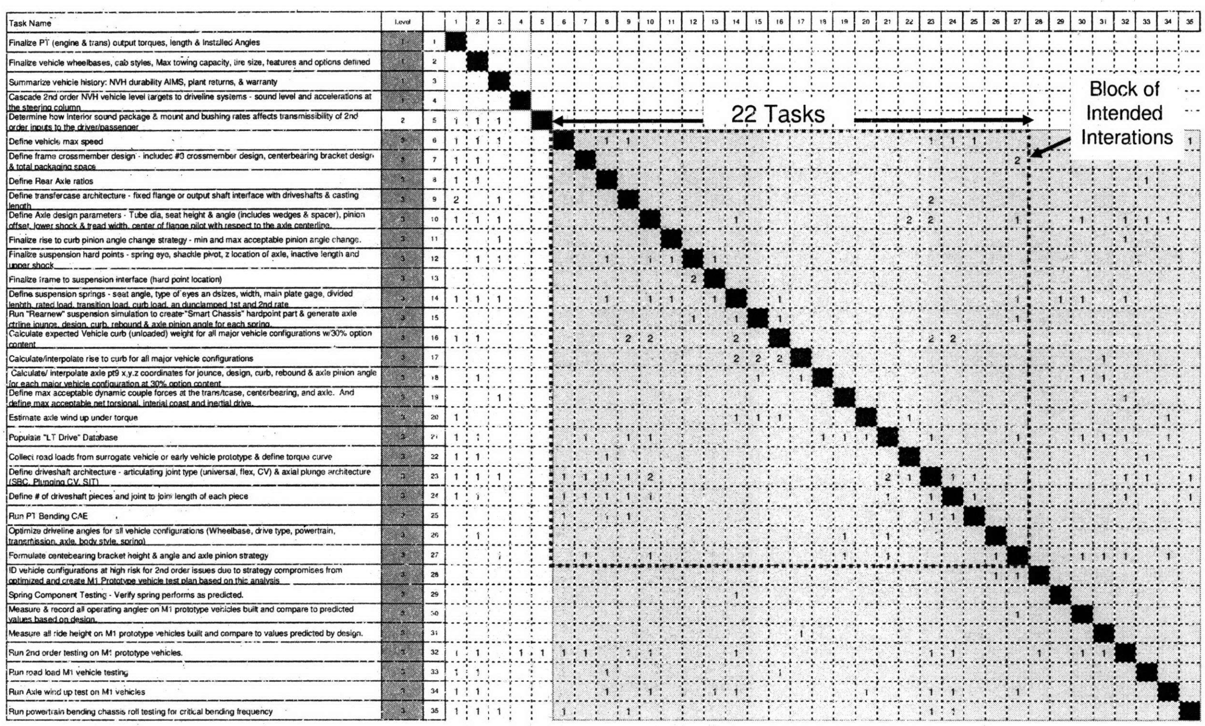


Figure 6.5: 35 Task DSM Partitioned with Block of Intended Design Iterations Shown

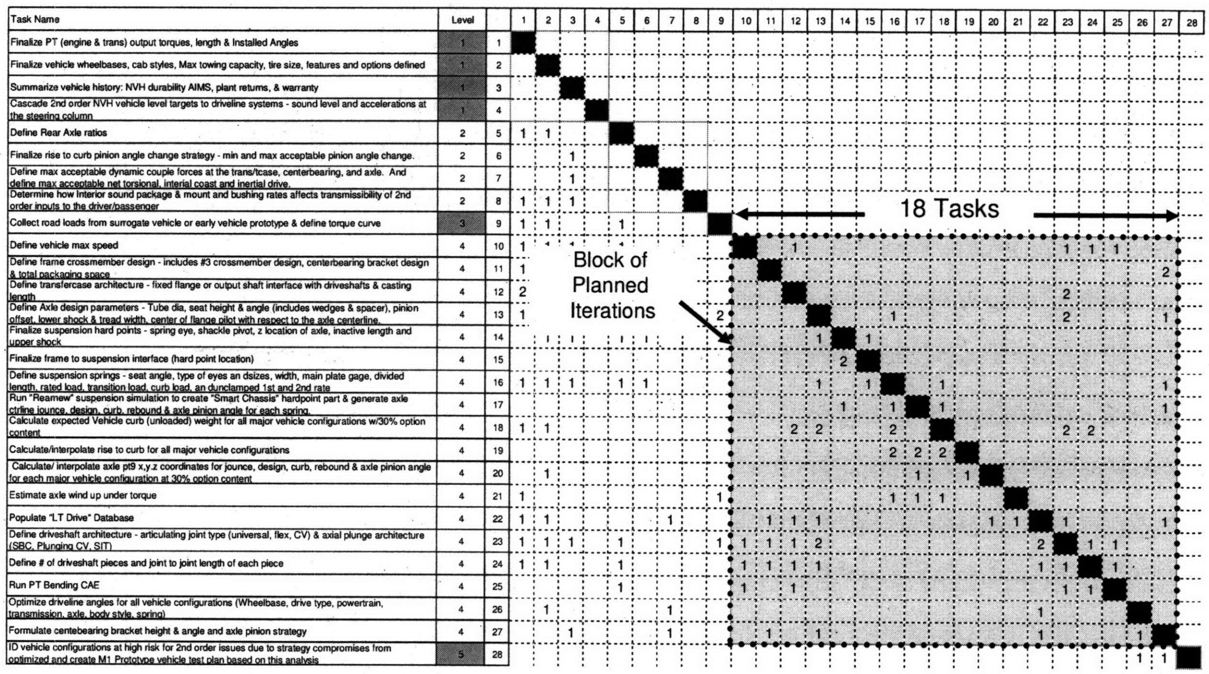


Figure 6.6: Partitioned DSM Excluding Unintended 1st VP Testing Iterations with Block of Intended Design Iterations Shown

As we see with the partitioning results above, partitioning the DSM, with the unintended iterations included, does not significantly reduce the "As-Is" DSM large iterative block. Thus, we conclude that partitioning will not yield a significantly improved process until the probability of occurrence for unintended iterations is drastically reduced. We will need to continue with the partitioned complete 35 task DSM and use additional process analysis and improvement tools that require a depth of knowledge in the technical aspects of 2nd order NVH as well as organization and process expertise.

6.5.2 Identifying Information Bottleneck & Critical Input Tasks in As-Is Partitioned DSM

We will start our process analysis by identifying the critical input tasks and information bottleneck tasks in the "As-Is" process DSM. Figure 6.7 identified the ten most critical input tasks to the current 2nd order NVH attribute management process. Each of these tasks feed information in the form of deliverables to the most upstream and downstream tasks in the process. Therefore, it is critical that the information produced in these "critical input tasks" is of a high degree of accuracy, delivered on time, and is rarely altered once completed. In figure 6.7, each critical input task is shown with its associated task variability and total number of input dependencies that have a high probability of rework. This enables us to see which critical input tasks are accurate and stable and which are not. This analysis flags the following four tasks as critical input tasks that also happen to a have high degree of variability, are likely to be changed late in the process, or are likely to yield results that prove earlier work to be in error and thus must be repeated.

- Define Rear Axle Ratios
- Define Rear Axle Design Parameters - Seat Angle & Shimming Strategy
- Complete 2nd Order Vehicle Prototype Testing
- Define Suspension Springs

The process restructuring team must determine what process changes can be made to stabilize the deliverables created in these four tasks and ensure the deliverables are always completed on time. Vehicle Prototype testing is flagged as a critical task with high variability. Thus, if the design fails this vehicle prototype testing many tasks must be repeated. So, in this case the goal would be to reduce the probability of this test finding that the design does not meet 2nd order attribute targets. If this test fails to meet expected criteria, a large amount of rework is created as indicated by this being called out as a critical task.

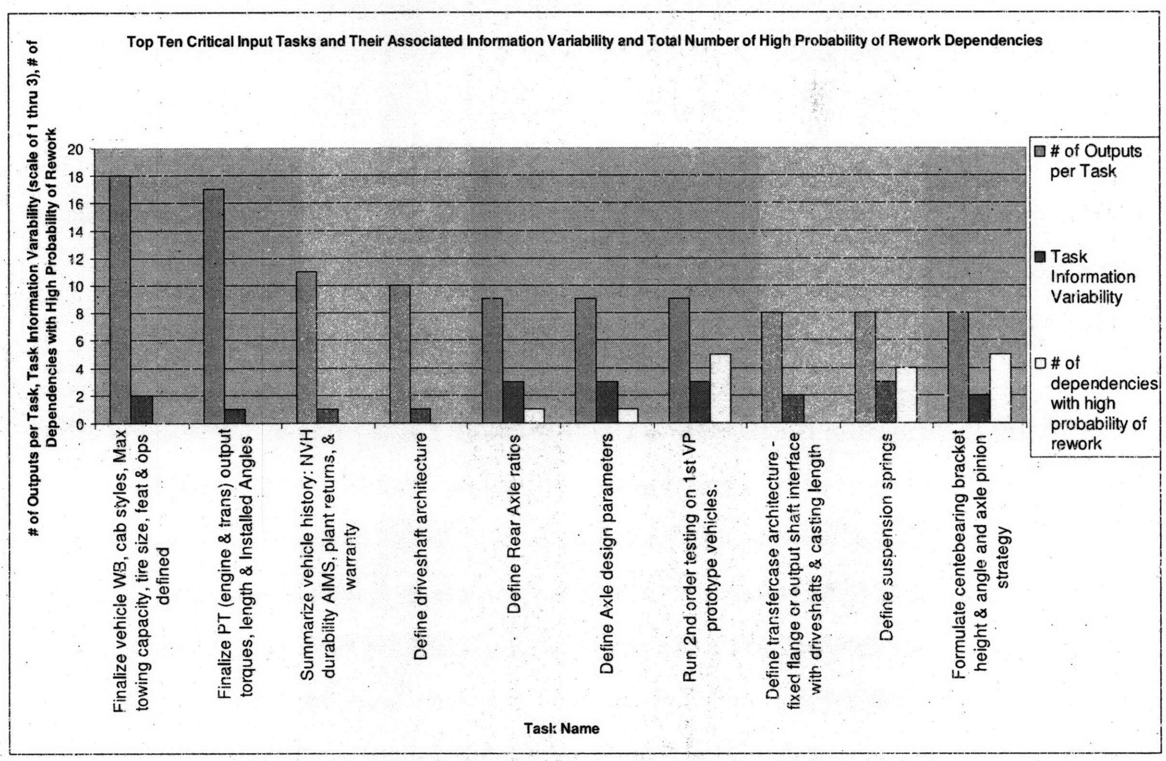


Figure 6.7: Top Ten Critical Input Tasks & Associated Information Variability & Number of High Probability of Rework Dependencies

Next we identify the information bottleneck tasks that exist in the current process. Figure 6.8 depicts the top ten task information bottlenecks along with the information variability and total number of high probability of rework dependencies associated with each task. Here we notice that there are many large information bottlenecks in this process. This means that there is a high potential for waiting if input tasks feeding these information bottlenecks are not completed at the correct time. The process restructuring team will need to create a strategy to stabilize inputs as well as ensure that input deliverables are completed at the right time. The information bottlenecks, tasks at highest risk of being delayed, in this process are:

- Define Driveshaft Architecture
- Populate LT Drive Database
- Run 2nd order 1st VP testing
- Define Suspension springs
- Define total number of driveshaft pieces and driveshaft lengths
- Define axle design parameters
- Define center bearing bracket height and axle pinion angle strategy

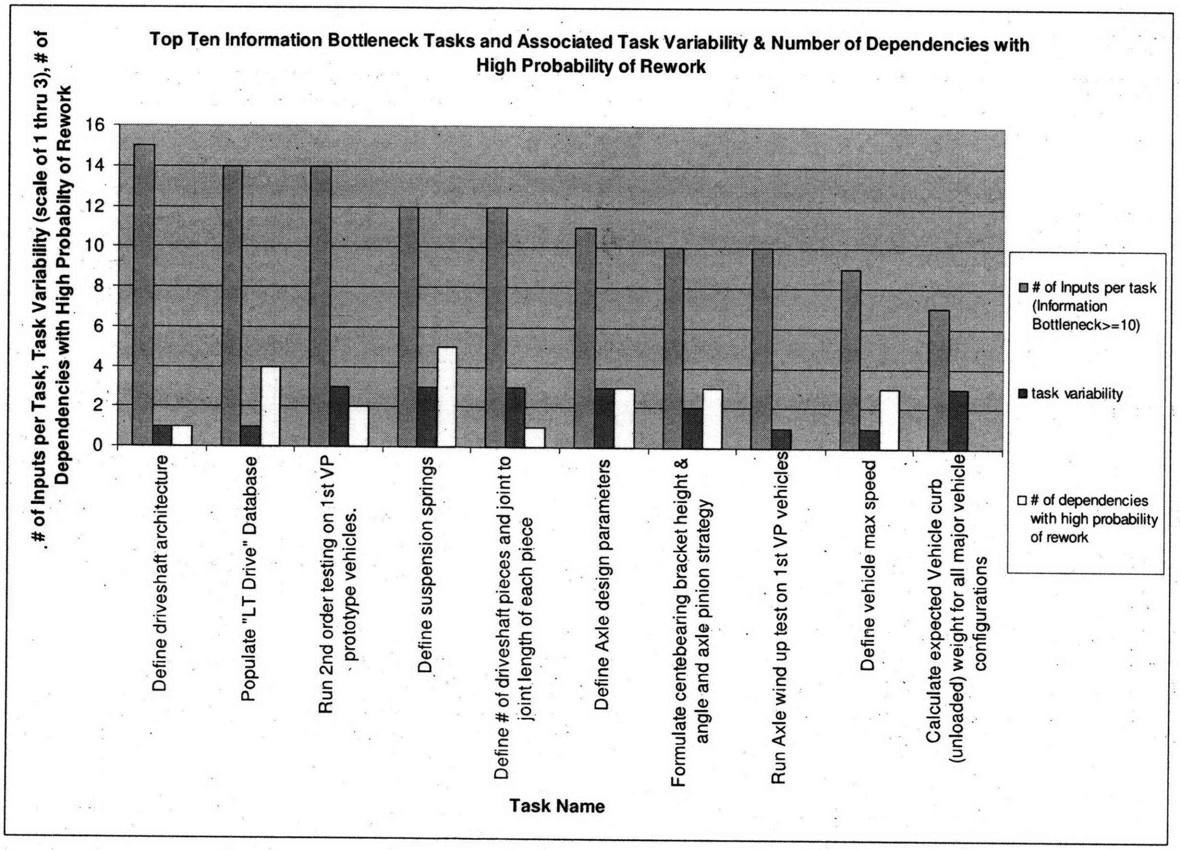


Figure 6.8: Top Ten Information Bottleneck Tasks and Associated Task Variability and Number of Dependencies with High Probability of Rework

It is important to call out tasks that are both a critical input task and an information bottleneck task. These tasks indicate the points in the process with the highest potential of delaying the entire process. The team must focus on how to both stabilize the information created by these tasks as well as ensure that inputs with little or no slack to these tasks are received at the correct point in time. These tasks are listed below:

- Define rear driveshaft architecture
- Define rear axle design parameters
- Complete 2nd order NVH 1st VP testing
- Define Suspension Springs
- Define center bearing bracket height and axle pinion angle strategy (define the universal joint cardan angles)

6.5.3 Process Restructuring

Efforts to reduce the single large iterative loop in this DSM to multiple smaller iterative loops, such as task re-sequencing and DSM partitioning have offered little improvement to the process. The process of designing the vehicle system to meet 2nd order NVH targets is inherently iterative and coupled. Thus, the process should be restructured using the "Do-it-right-the-first-time" (DRFT) method [Yassine et al., 2000]. As discussed in chapter five, the DRFT method of DSM process restructuring involves redefining existing tasks, adding new tasks, and deleting tasks, all with the goal of reducing uncertainty early in the process to avoid iterations late in the process. 2nd order design and development process time can be reduced as well as product quality be improved by: 1) reducing the total number of iterations by reducing the probability of rework, 2) reducing the total task time by making tasks easier to execute, 3) increasing the iterative learning curve, meaning once the work is done making successive iterations shorter, or 4) overlapping existing tasks and implementing concurrent engineering where appropriate.

6.5.3.1 Design, Build Vehicle Prototype, Test, Redesign Loops

The DSM reveals two large unintended iterative loops where there is a high probability that failed vehicle level testing will cause the design cycle to iterate. The two vehicle prototype test tasks that have a high probability of rework are:

1. Measure & record all operating angles on from 1st VP build and compare to predicted values.

2. 2^{nd} Order NVH vehicle prototype testing

Discussion on unintended iterative loop #1: As stated earlier, 1^{st} VP is intended to have a production representative underbody design. At the 1^{st} VP build, driveline angles are measured on all vehicles built and these angles are compared to the design intent angles. Interviews have revealed that for most programs, the vehicle prototype driveline angles deviate unexpectedly from the intended values. This is due to the fact that the nominal location of subsystems that control the driveline angles are likely to not match intended location. In addition, manufacturing variation in controlling subsystems is not always well understood. The powertrain "as installed" angle is often lower than represented in CAD models. The rear suspension springs and rear axle pinion angle are often not located as predicted by analytical tools. This means that the teams need to reassess their subsystem design. Often the axle pinion angle and driveshaft center bearing attach to frame height strategy has to be redeveloped at this point in time. Some process solutions to reduce the probability that 1^{st} VP driveline angles deviate from expected driveline angles are listed in table 6.3.

Discussion on Unintended Iterative Loop #2: The current 2^{nd} order NVH attribute management process relies heavily on redesign after physical vehicle level prototype testing. It is a bit obvious to state that an increase in early analytical work is needed. However, in the case of 2^{nd} order NVH, the driveline systems engineers are the first to point out this fact. The question is how should the process be changed to produce earlier design decisions with a higher degree of confidence and with what early analytical tools? How can the process be improved to yield a lower probability of rework after 1^{st} VP 2^{nd} order NVH testing? These questions are answered in table 6.3 and discussion in section 6.5.5.

6.5.3.2 Large Planned Iteration Loop Process Improvement

After significantly reducing the probability of rework for unintended iterations, the team focused on how to improve the intended iterations that occur as a part of the expected design process. Currently, the process to design subsystems such that the vehicle system meets 2^{nd} order NVH targets is a coupled and iterative cross-functional design challenge. The team studied the two largest planned iterative loops; 1) Design the driveline angles under all vehicle weight and driveline torque conditions and 2) Design the driveline subsystems such that the vehicle can reach maximum speeds and not approach driveline resonant frequencies. The team proposed several process restructuring actions that reduce task time and probability of rework. These are summarized in tables 6.3 and 6.4.

6.5.4 The Process Restructuring Proposal

By analyzing the "As-Is" DSM, the 2nd order attribute management process improvement team identified many opportunities for improved product design, project management, and cross organizational communication. These process improvements include early and accurate product assumptions, system and subsystem level design rules, standardized interface architecture, a choice of off-the-shelf hardware solutions, the use of simple desktop analytical tools as well as more advanced CAE tools, and finally the addition of hardware bench testing prior to the 1st VP build.

The process improvement proposals have been divided into two categories; 1) improvements that do not require any capital investment and 2) improvements that do require capital investment to implement. These two categories were chosen to show management the effects of process changes that could be implemented immediately with no investment vs. those process improvements that require an investment. Management can quantitatively discern the difference in relative process lead time and standard deviation improvement between those actions that require investment and those that do not. It is important however, for management not to base process improvement decisions on quantitative change in lead time and variance alone. The goal is not only to decrease development time, but to decrease required project resources and improve product quality. In this case, product quality can be measured in terms of adherence to 2nd order NVH targets. All quantitative lead time improvements should also be presented with assessment on predicted product quality and project resources impact. Some process improvement proposals do not reduce the process lead time or standard deviation. However, the action is predicted to improve product quality or reduce required resources. Table 6.3 and Table 6.4 list the process improvement proposals and their effect on the DSM and the change in process lead time and associated standard deviation and coefficient of variation.

#	Classification	DSM Task Type	Process Change Description	Effect
1	Early Assumption	New Task	Create Early LT Drive Database for initial d/s design & 2^{nd} order NVH analysis. Use mix of carryover from previous program & early program suspension data to populate Database.	↓ iterative loop size ↓ rework probability
2	Improved X-Functional Communication	Modified Task	Develop system level design guide for ideal rear suspension defined rear axle pinion travel & angle change.	↓ task time ↓ rework probability
3	Improved X-Functional Communication	New Task	Driveline Systems, Frame, PT mounts & PT meeting to establish PT installed angle behavior.	↓ rework probability
4	Early Assumption	Modified Task	Start with 2nd order force bogies from previous program updated with any new vehicle sensitivity assumptions.	↓ rework probability
5	Improved X-Functional Communication	New Task	Create simple program specific x-functional 2^{nd} order Deliverable Gantt chart for critical deliverables.	↓ rework probability
6	Simple Desktop Analytical Tool	Modified Task	Driveshaft engineers to use simple desktop PT Bending spreadsheet calculators before sending design to CAE group.	↓ rework probability
7	Early Assumption	New Task	Driveline Systems negotiate acceptable complexity w/frame, rear axle & B&A VO	↓ rework probability
8	Concurrent Engineering	Modified Tasks	Ensure that Driveshaft Design and Suspension Design is being designed concurrently early in the product development cycle.	↓ process time
9	Task Overlap	Modified Tasks	Overlap all task dependencies identified as having fast upstream evolution & low downstream sensitivity	↓ process time
10	Improved Design Space	Modified Task	Driveshaft and frame to negotiate frame #3 cross member design for best package space.	↓ rework probability
11	Improved Design Space	Modified task/New task	Delay #3 frame cross-member center bearing bracket height & angle definition as long as allowed by frame supplier w/o driving extra cost into the product.	↓ rework probability
12	Improved Design Space	Modified task/ New task	Delay rear axle pinion spacer/shim angle definition as long as allowed by rear axle pinion supplier	↓ rework probability
			Process Lead Time Simulation Results (% delta from As-Is DSM)	% $\delta\mu$ =13% % $\delta\sigma$ =22% % δC_υ =10%

Table 6.3: Process Restructuring Proposals – No Capital Investment Required

#	Classification	DSM Task Type	Process Change Description	Effect
1	Off-the-Shelf Hardware Solutions	Modified Task	Develop off-the-shelf driveshaft hardware solutions with supplier.	↓ task time ↓ rework probability
2	Standardize Interface Architecture	Modified task	Standardize transmission/ transfer case to driveshaft interface.	↓ task time ↓ rework probability
3	Desktop Analytical Tool	Modified Task	Write macro to automate LT Drive angle optimization exercise	↓ task time
4	Improved Analytical Tool	New Task	Develop vehicle CAE model to include predictive capability of response to 2nd order input forces.	↓ rework probability
5	New Bench/Rig Testing	New Task	Adapt current driveline rig to simulate road loads so that driveline angle dynamics & 2nd order response can be understood under high torque applications.	↓ rework probability
Process Lead Time Simulation Results (% delta from "to-be w/no capital investment" DSM)				% $\delta\mu$ =18% % $\delta\sigma$ =30% % δC_v =14%

Table 6.4: Process Restructuring Proposals – Capital Investment Required

6.5.5 Predicted Impact on Process Lead Time, Project Resources, and Product Quality

6.5.5.1 "To-Be" with No Capital Investment

Six completely new tasks were added to the process in the "To be with no capital investment" DSM. Adding tasks is counter intuitive to reducing process lead time. However, these added tasks cut down the probability of rework and increase the quality of the entire system. The new tasks include; several cross functional meetings early in the program to determine the feasible design space, communication with VO directly early in program and the use of simple desk top analytical tools prior to CAE evaluation.

In the new "To-Be" process, early cross organization communication is formalized in the form of scheduled meetings prior to early New PDS milestones. These meetings are formalized in New PDS local processes for the affected subsystems. The purpose of these meetings is to establish the available design space for design parameters affecting 2nd order NVH. At these meetings driveline systems engineers can establish the acceptable complexity for frame, rear axle and vehicle operations. These meetings increase product quality by giving driveline systems engineers more design space to optimize driveline angles and meet 2nd order NVH targets. These meetings reduce the total lead time by cutting down on probability of rework.

The use of simple desktop driveshaft bending analytical tools instead of costly CAE resources can bring significant process improvement. This research has found that these desktop analytical tools exist at NA OEM. However, the driveshaft engineers interviewed were not using this tool. In the current process, driveshaft engineers send design proposals directly to CAE to verify that the design meets PT bending. In the new process, driveshaft engineers use powertrain bending spreadsheets to see if their design meets PT bending before sending to CAE. CAE is very resource intensive. In addition, long queues form for CAE resources because the resource is over utilized by all engineering teams. Therefore, design engineers should be fairly confident in their design before handing the design off to CAE.

The decrease in process average lead time and decrease in standard deviation is also due to making several early assumptions. These early assumptions have the effect of tearing DSM dependencies that cause large feedback loops. One major early assumption in the new process is the creation of an initial database to populate a driveline angles and 2nd order force analysis tool (The application is referred to as Light Truck Drive or LT Drive in the DSM) early in the program. Creating an initial LT Drive database with assumptions about 2nd order maximum acceptable forces, axle wind up due to spring design, and suspension travel enables the driveshaft to be designed early in the program. Only minor changes need to be made to the driveshaft design later in the program when actual data is collected on the design parameters for which we made initial assumptions. These assumptions and the creation of an early LT Drive analysis tool database acts to decouple the driveshaft and spring design early in the program. Instead of these two design process occurring sequentially they can occur in parallel. Then, once the spring design is finalized only minor changes may be needed for the driveshaft design.

Another major improvement in process lead time, quality and resources is achieved by the creation of system level design rules. In the new "to-be" process suspension design starts with input from driveline in the form of a system level design rule that states the required axle pinion travel pattern to enable optimal 2nd order NVH performance. The suspension hard points and spring design dictate how the rear axle pinion will travel as the suspension moves between full jounce and full rebound. The driveline engineers will have input into these suspension design parameters where the driveline engineers previously assumed that axle pinion travel patterns was a pure input.

The new to-be DSM created by process improvements that do not require any capital investment is shown in figure 6.9. The process, resource and product quality improvements were mostly achieved by process restructuring with the addition of new tasks, deletion of some old tasks, and modification of many tasks.

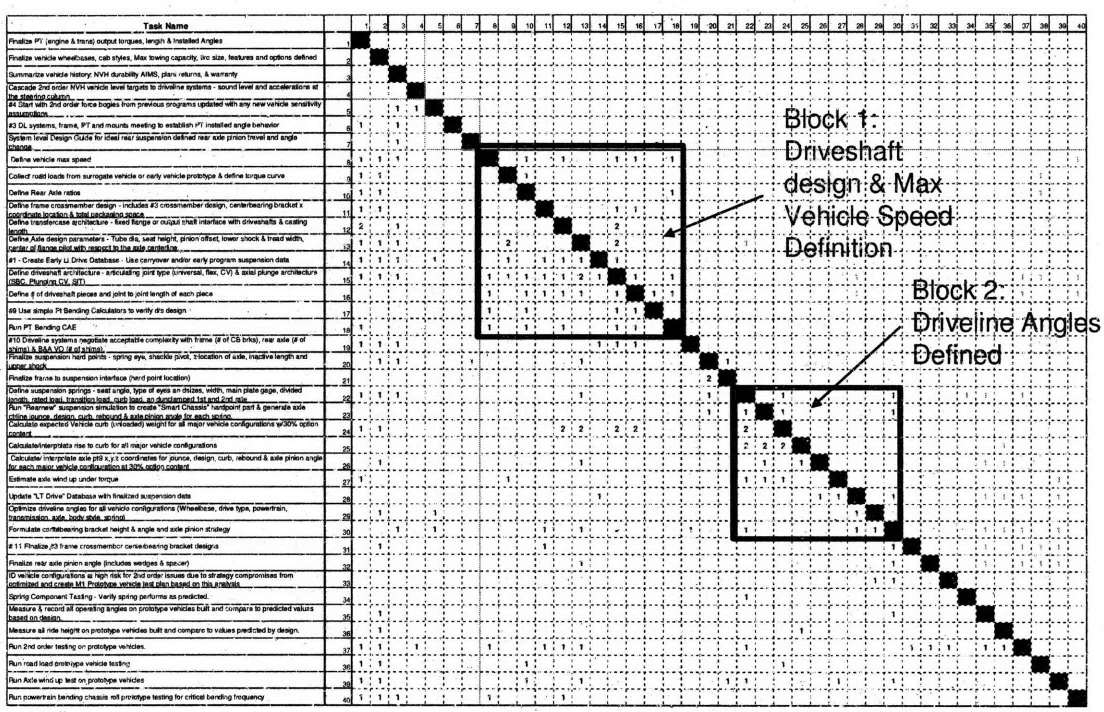


Figure 6.9: "To-Be" with No Capital Investment DSM Structure

The "to be with no capital investment" process was simulated using the excel macro discussed earlier in this chapter. Figure 6.10 represents a histogram summarizing the "to-be with no capital investment" results of a 1000 run Monte Carlo simulation. Using the "As-Is" DSM simulation results as the baseline, we are able to achieve approximately 13 percent improvement in the mean process lead time, 22 percent improvement in process standard deviation, and ten percent improvement in the coefficient of variation for the process standard deviation and mean. These results support the organizational effort to implement all of the process changes outlined in table 6.3 and listed in the new "to-be with no capital investment" DSM.

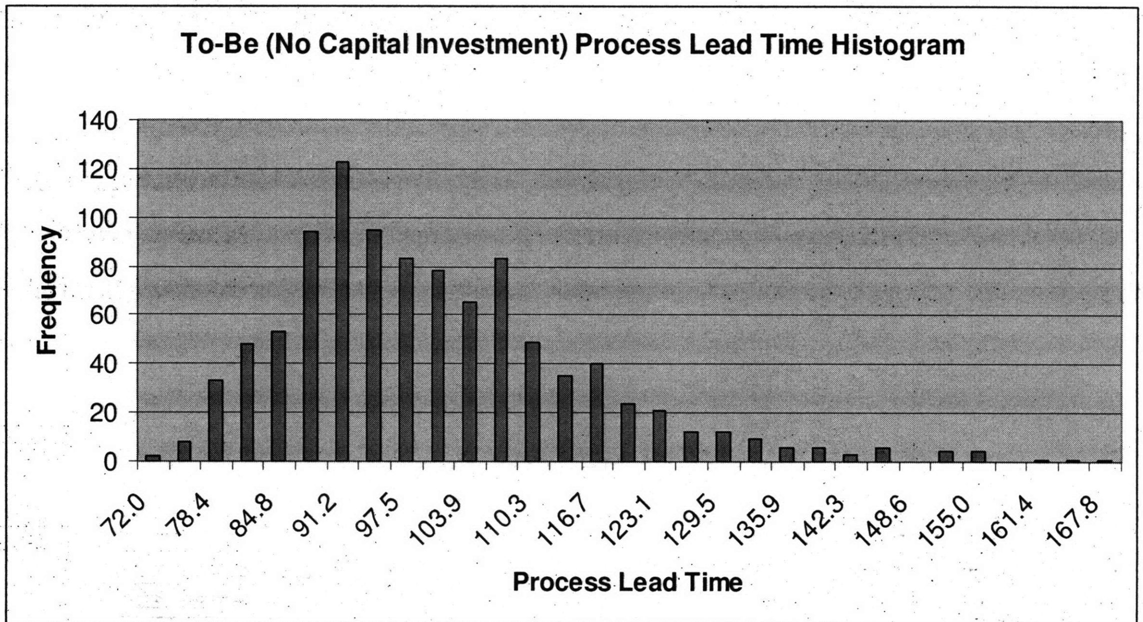


Figure 6.10: To-Be With No Capital Investment Process Lead Time Histogram

6.5.5.2 "To-Be with Capital Investment"

Additional improvements in process lead time, resource requirement, and product quality can be gained with some capital investment in expanded CAE capability, purchase of improved desktop analytical tools written by outside consultants, development of off-the-shelf hardware solutions with suppliers, and the expanding the capabilities of the current driveline NVH rig to include 2nd order NVH data collection. The improvements are summarized in table 6.4 above.

In the "As-Is" DSM, the task "Formulate center bearing bracket height & angle and axle pinion strategy" is a critical input. The output from this task feeds design parameter decisions for axles, suspension, frame, driveshaft and powertrain. The process lead time to formulate the strategy can be drastically reduced with the creation of an LT Drive application macro that automates the "driveline angle optimization" task. Also, the strategy creation task time can be drastically reduced by expanding LT Drive to include automated multi-attribute trade-off analysis capability. Multi-attribute trade-off analysis will be discussed in chapter seven.

After leveraging desktop analysis tools, such as LT Drive, early in the design process, the capability of current vehicle level NVH CAE tools can be expanded to include 2nd order NVH analysis. This would enable vehicle engineering to cascade realistic 2nd order NVH targets to driveline, the owner of this attribute. Currently, in the "As-Is" process, generic maximum 2nd order force targets are used in the development of the driveline, suspension, powertrain, and frame driveline parameters that control 2nd order NVH. The CAE vehicle model used for NVH analysis could predict the likely vehicle sensitivity to 2nd order inputs. This CAE vehicle model would improve attribute management across organizational lines as well as target cascading.

Finally, significant process improvement can be achieved by adding a hardware rig test phase. This hardware rig test would occur on the current driveline NVH rig which would have to be modified to achieve 2nd order NVH test and analysis capabilities. Adding this 2nd order NVH rig test phase may seem counter intuitive to reducing process lead time because the predicted lead time to set up, run the rig test, and analyze the results is significant. However, the main effect of the rig testing is to drastically reduce the probability of rework at 1st VP 2nd order NVH testing.

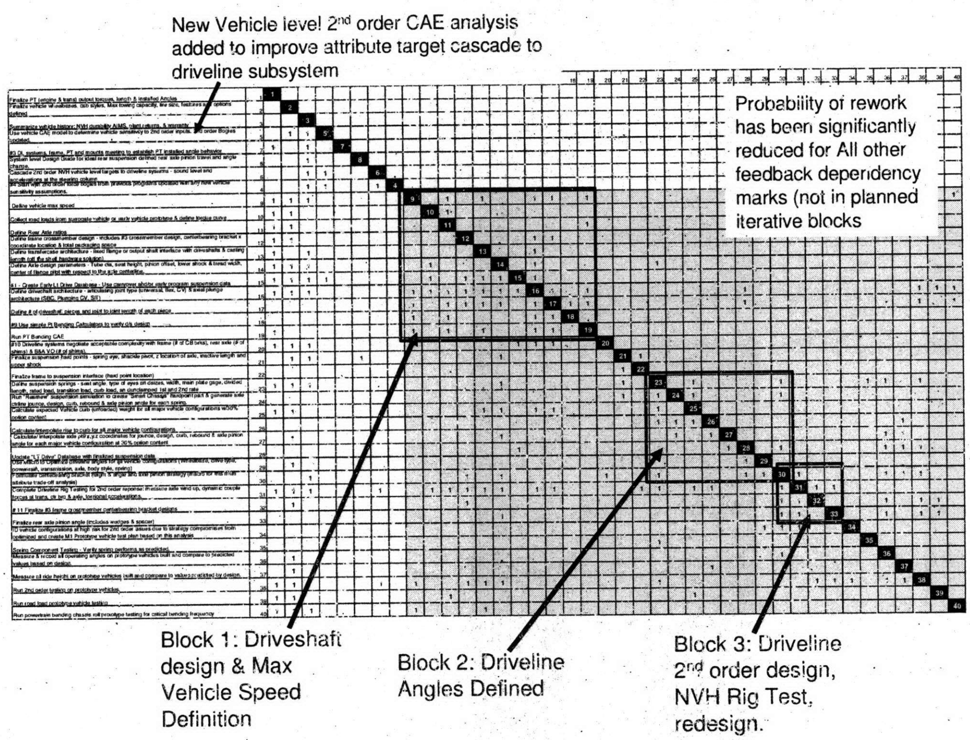


Figure 6.11: To-Be with Capital Investment Process DSM

The "to be with capital investment" process was simulated using the excel macro discussed earlier in this chapter. Figure 6.12 represents a histogram summarizing the "to-be with capital investment" results of a 1000 run Monte Carlo simulation. Using the "to be with NO capital investment" DSM simulation results as the baseline, we are able to achieve approximately 18 percent improvement in the mean process lead time, 30 percent improvement in process standard deviation, and 14 percent improvement in the coefficient of variation for the process standard deviation and mean. These results support the potential cost to implement all of the process changes outlined in Table 6.4 and listed in the new "to-be with capital investment" DSM.

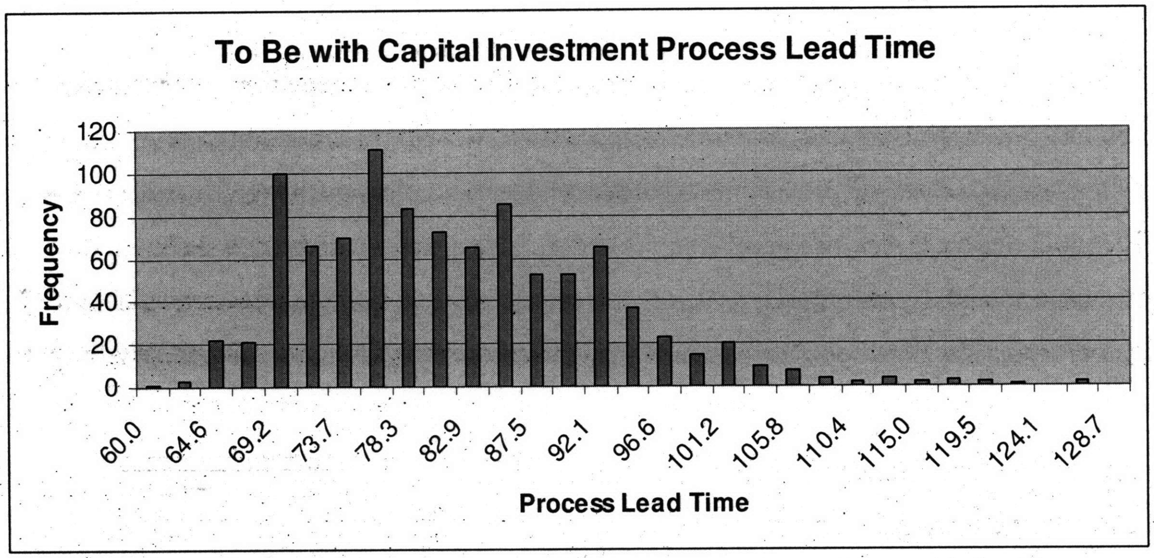


Figure 6.12: To-Be With Capital Investment Process Lead Time Histogram

6.6 Chapter Summary

The DSM method for process modeling proved to be a valuable tool for improving the total time to develop, integrate and test subsystem design to meet vehicle level attributes that are affected by many different subsystems. The application of the DSM methods for process modeling and improvement was able to create a predicted 30% reduction in average engineering time dedicated to meeting 2nd order NVH attribute performance targets and a 45% reduction in lead time variation. However, the DSM creation process is time consuming and required expertise in the DSM method itself as well as in the process under study. Therefore, the DSM method for process modeling is recommended only for those vehicle level attributes deemed as "high impact." Again "high impact" attributes are defined as attributes that have caused significant launch issues in the past or are causing high incidence of warranty claims.

Figure 6.13 shows a high level flow diagram of the "To-Be" process. This flow diagram omits more detailed tasks from the DSM to relay the flow of information in the new proposed process. The black arrows represent feed forward information flow. The red arrows represent feed back information flow. The thickness of the red feed back arrows is used to represent the relative probability of rework for these feed back loops. Looking at this figure we see that there are no large iterative loops with a significant probability of rework. Large iterative loops have been "torn" by the addition of early cross functional communication meetings and early design assumptions. An example of a large iterative loop is the driveshaft design loop. Early assumptions about suspension travel and 2nd order vehicle sensitivity enable the creation of an early "2nd Order NVH Force Analytical Tool Database." This early and accurate database of suspension travel and 2nd order force predictions enables a high confidence driveshaft design earlier in the program. Also, frequent iteration between driveshaft and powertrain bending CAE has been reduced with the use of desktop powertrain bending analytical tools. The probability of rework at 1st VP testing has also been significantly reduced with the addition of driveline 2nd order NVH rig testing. Note that a more detailed account of the process changes and effect on rework loops and rework probability can be found by studying the "To-Be" Process DSM presented in figure 6.11.

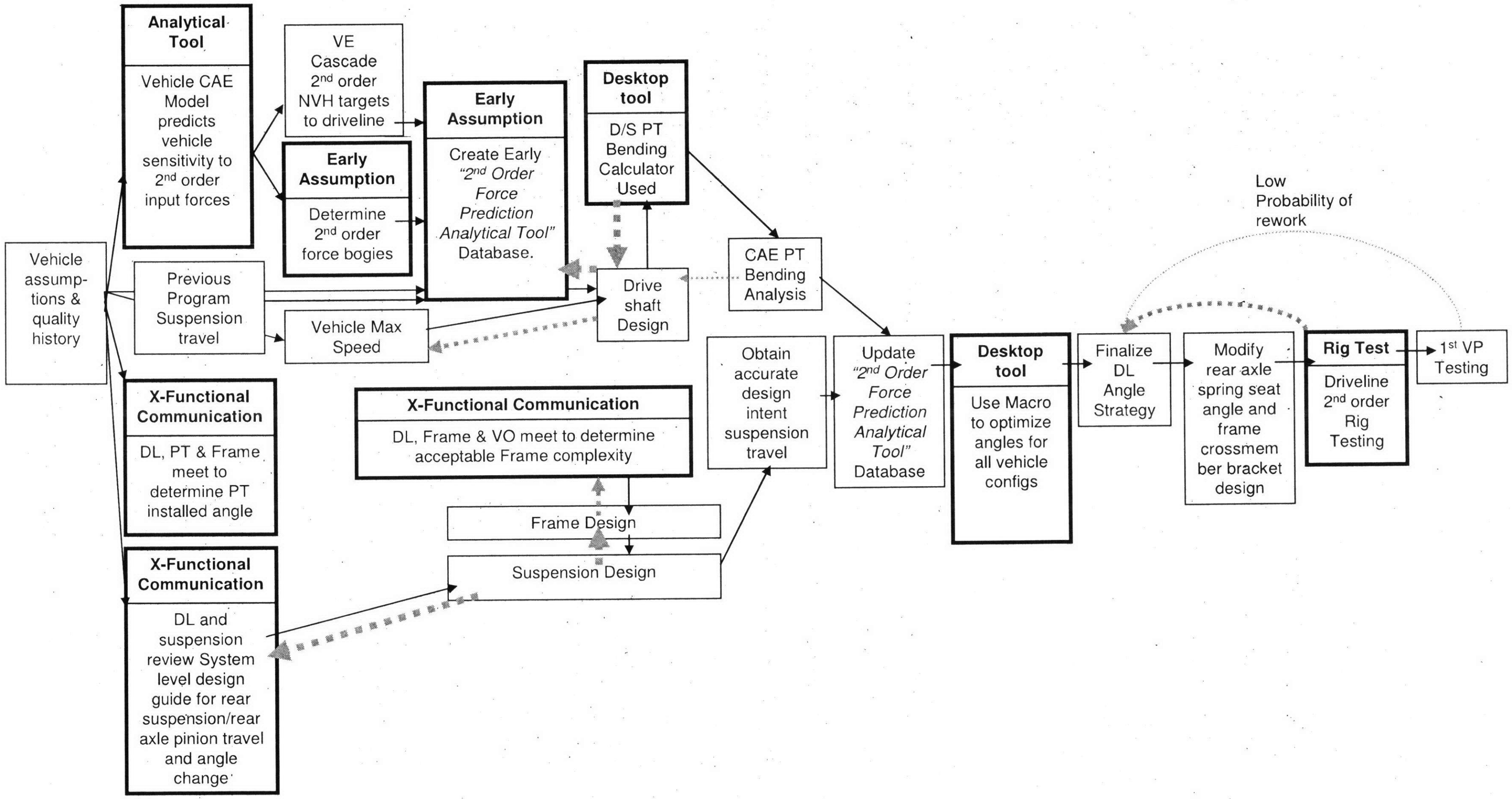


Figure 6.13: High Level Process Flow Diagram of New "To-Be" 2nd Order NVH Management Process

7. ATTRIBUTE OWNERSHIP & CONTROL, TRADE-OFF AND CASCADE

In this section, a high level overview of current attribute target ownership and control, trade-off processes, decomposition and cascading at NA OEM is introduced. Current attribute management processes, organizational and cultural issues are discussed. A systems engineering approach to attributes ownership and control, trade-off, decomposition and cascade is introduced. The implementation of a knowledge based tool for documenting attribute interactions is explored. The research in this chapter segues to future research on the topic of system level attribute trade-off, decomposition and cascade within various product development organizations and the potential effect of a systems engineering approach on attribute trade-off, decomposition and cascade.

7.1. Attribute Types

Different vehicle level attributes are affected by different numbers of first order subsystems and design parameters, designed by engineers in different organizations. Consequently, attribute management approaches must be tailored to the attribute. For the sake of this thesis, we distinguish between three types of vehicle level attributes. These three types of attributes are depicted in figure 7.1.

We define "high level" attributes as those attributes that involve most or all major subsystems. Examples of "high level" attributes are cost, weight, fuel economy, NVH, vehicle dynamics, durability, and safety. Some "high level" attributes such as cost, weight and durability can be "decomposed" hierarchically from system, to subsystem and component levels because they truly are the "sum of its parts." Other "high level" attributes such as NVH and vehicle dynamics cannot be simply decomposed hierarchically because the subsystem designs that deliver these attributes are extremely coupled. "High level" attributes are owned at the vehicle level by vehicle engineers and managed by Program Attribute Teams that fall under the vehicle engineering organization. Nevertheless, their achievement involves many engineers in many organizations and requires active management.

We define "mid level" attributes as those whose response is dependent on design parameters controlled by three or more major subsystems. Management of these attributes is of particular interest because these attributes are significantly cross functional but are sometimes owned at the subsystem level. 2nd order NVH, the focus of this thesis, can be considered a "mid level" vehicle attribute being owned by driveline but influenced by design parameters controlled by powertrain, transmission, frame, and suspension. A principal aim of the DSM study in Chapter 6 is to improve management of this kind of attribute.

We define "low level" vehicle attributes as those whose response is dependent on design parameters controlled by only one or two major vehicle subsystems. These vehicle attributes are always owned at the subsystem level. Examples of "low level" vehicle attributes are seat comfort, dash panel appearance and brake noise. There are hundreds of these "low level" vehicle attributes. These attributes are usually managed by subsystem integration engineers. Figure 7.1 shows all of the major vehicle subsystems and gives examples of high, mid and low level attributes.

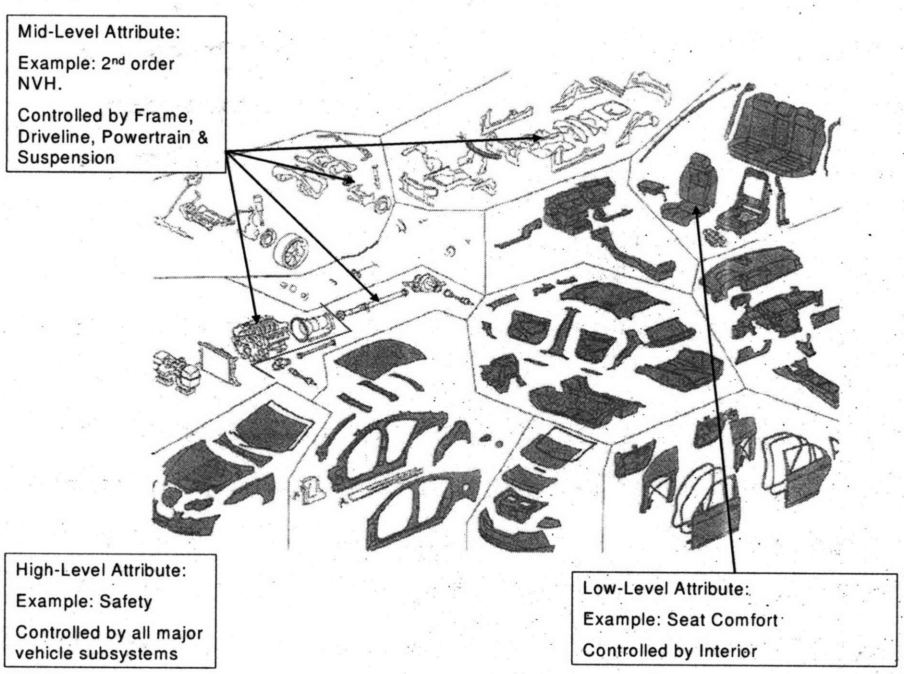


Figure 7.1: The Major Vehicle Subsystems and Three Types of Attributes Depicted

7.2. Attributes Management at NA OEM

7.2.1 Attribute Ownership and Control at NA OEM

At NA OEM, high level attributes are owned by Program Attribute Teams (PATs). This ownership means that the PATs are responsible for ensuring that attribute targets are met. Mid level attributes are owned either at the vehicle or subsystem level depending upon the nature and impact of the attribute. Low level attributes are always owned at the subsystem organization level. A PAT is assigned to each individual high level vehicle attribute. The PATs are part of the vehicle engineering (VE) organization at NA OEM. Each PAT is assigned a PAT leader who takes on the leadership role for ensuring the PAT delivers its attribute targets while being cognizant of trade-offs with other attributes. The PATs own the attribute, yet do not control any of the subsystem designs. Thus, it is critical for PAT engineers to build strong working relationships with the subsystem organizations that control their attribute. Within the NA OEM PD organization, the PATs and the subsystem engineers are under completely different management chains that are only joined at the vice president level. Thus, attribute success depends on strong lines of communication across reporting chains. Close proximity, geographically, between the PAT and the subsystems that control their attribute provides an advantage as face to face meetings are imperative for the current attributes management process. The details of this process will be discussed later in this chapter. A list of the high level attributes that are assigned to PATs is shown in Table 7.1. The PATs are responsible for, and thus own, performance delivery of these high level vehicle attributes.

"High Level" Vehicle Attributes

	Design	TASE	Aerodynamics
	Manufacturing	TASE	Heat Management
	Perceived Quality/Craftsmanship	Trailer Tow	Payload
	Performance Feel	Trailer Tow	Towing Dynamics
	Shift Quality		Off Road
NVH	Powertrain NVH		Service/Accessories
NVH	Interior (Body/Chassis) NVH		Service Cost of Ownership
NVH	Windnoise		Package
NVH	Squeak & Rattle		Ergonomics
	Vehicle Durability		Seat Comfort
Vehicle Dynamics	Ride		Climate Control
Vehicle Dynamics	Handling / Traction		Safety
Vehicle Dynamics	Steering		Security
	Brakes		Electrical /Electronic Features
	Fuel Economy		VI Core Attributes
	Emissions		
	Powertrain Cooling		

Table 7.1: List of High Level Vehicle Attributes

As discussed previously, the mid and low level attributes are usually owned at the subsystem level but are often controlled by more than one subsystem. Thus, the subsystem that owns the attribute has the responsibility to maintain strong lines of communication with all of the controlling subsystems. As in the example of 2nd order NVH, most of the controlling subsystems fall under different management chains. Disputes with attribute and design trade-offs often occur between the organizations that control and the organizations that own various attributes. Thus, attribute prioritization at the mid and low level is imperative to successful trade-off and management of these attributes. Approaches to attribute prioritization and trade-off of these mid and low level attributes will be discussed later in this chapter.

7.2.2 High Level Attribute Trade-off Process at NA OEM

At the beginning of a new vehicle program, an overall attribute priority rating is assigned to each high level attribute. This attribute priority rating is based on a program attribute leadership strategy devised by marketing and benchmarking research. In this program strategy, each attribute is categorized as being "leadership," "among the leaders," "competitive," or "uncompetitive" for that particular vehicle program. A "leadership" attribute priority rating means the attribute should set the vehicle brand apart from other competitors. On the other hand, those attributes categorized as "uncompetitive" are not relevant to supporting the vehicle brand and should be met at the lowest possible cost. This priority rating is intended to facilitate trade-offs between attributes. The PAT leader is responsible for setting attribute targets that achieve the program attribute leadership strategy. However, these attribute targets must be compatible with other attribute targets as well as with quality, cost, weight and functional targets. The PAT leader is also responsible for developing plans for the tests required to verify attribute performance.

New PDS adopted attribute management methodologies that have been proven successful at an overseas partner OEM. Surprisingly, these methodologies do not include complex technical trade-off tools such as "Multi-Attribute Trade-Off Analysis" and "multi-variable system optimization," methods that will be discussed later in this chapter. A New PDS Attributes Expert stated that technical attribute trade-off analytical tools such as those used in the aerospace and defense industry were once implemented at NA OEM but were proven unsuccessful. The failure of these tools was due in part to poor engineering confidence in the pure numerical quantitative results. An NVH PAT leader familiar with the failed attempt at a formalized computational system for high level attribute trade-off said that the system was "a disaster." He stated that this system attempted to trade-off and optimize too many parameters at one time. He said that some of the inputs into the system were low quality due to inexperienced engineers and due to engineers "playing" the system. Engineers could falsely weight their attributes leading to faulty trade-off results. He said that the output of the system was rarely trusted because there was not a good understanding for the output results. All PAT engineers interviewed expressed disapproval of any pure analytical trade-off tools for high level attributes that involve hundreds of design parameters. One engineer stated that with these tools, a deep understanding of the system and how it works is often lost as engineers try to rely on an analytical model to make decisions for them. Thus, the tool was abandoned by engineers for high level attribute trade-off analysis and subsystem design selection. As discussed later in this chapter, these analytical tools could potentially provide value to some mid and low level attribute trade-offs.

The current attribute trade-off methods that New PDS uses are focused on high level attributes only. A flowchart of the process is shown in Figure 7.2. Each of the tools named in this figure is described in the following paragraphs. There are no specific formalized processes in New PDS for managing mid and low level attribute trade-offs owned at the subsystem level.

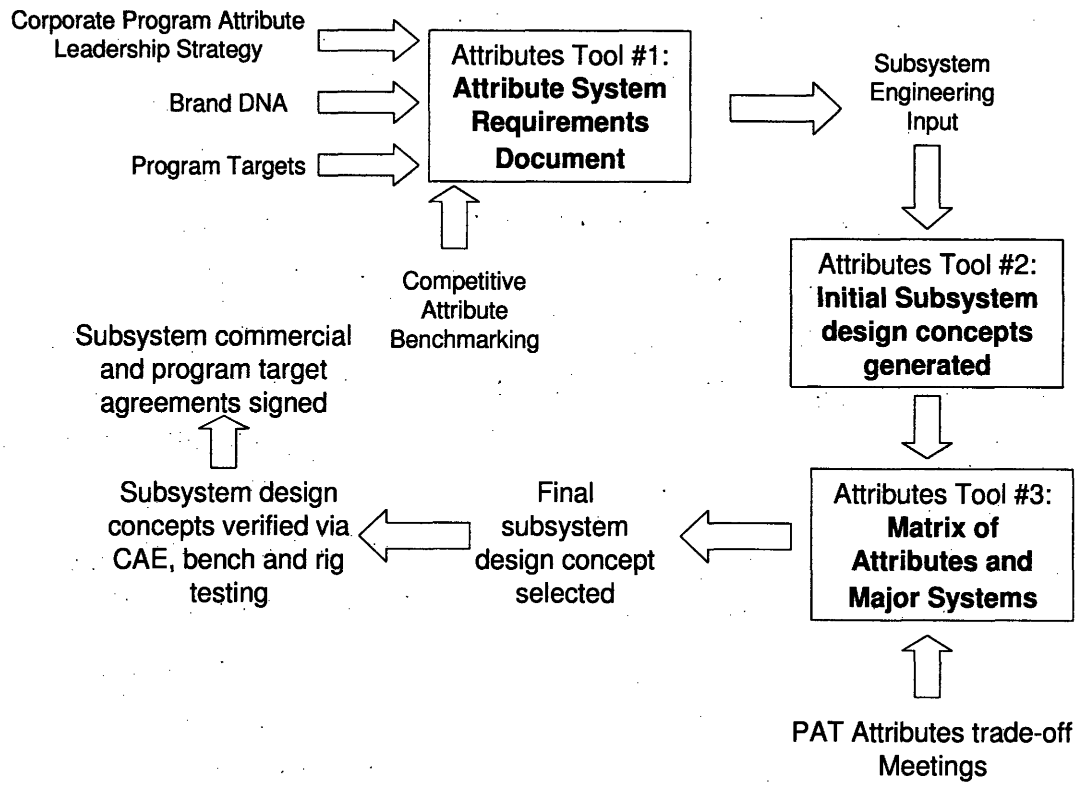


Figure 7.2: High Level Attributes Management Process led by PAT

The New PDS method for high level attribute trade-off management is based on subsystem design space exploration followed by controlled concept convergence assisted by spreadsheets that aim to align subsystem design concepts with attribute and functional targets. The New PDS process used at NA OEM mirrors the "Pugh Method" or "Decision-Matrix" method for concept generation and convergence. The Pugh Method is a framework for engineering design concept selection. This method requires that a team produce as many design concepts as possible. The total number of concepts is then narrowed, expanded again, and then narrowed until the team converges on the best possible design solution. The best design solution is often a hybrid of several original ideas, thus emphasizing the importance of the iterative expanding and narrowing of the list of design concepts. The Pugh Method analyzes and then narrows the list of design concepts by use of a "Pugh Matrix." The matrix lists all concepts against system requirements and makes quantitative comparisons of each concept to a chosen baseline design. Each concept is rated against the baseline design for each system requirement. Figure 7.3 depicts controlled concept convergence process. Figure 7.4 is an example of a decision matrix used to quantitatively assess each design concept. [Pugh, 1991]

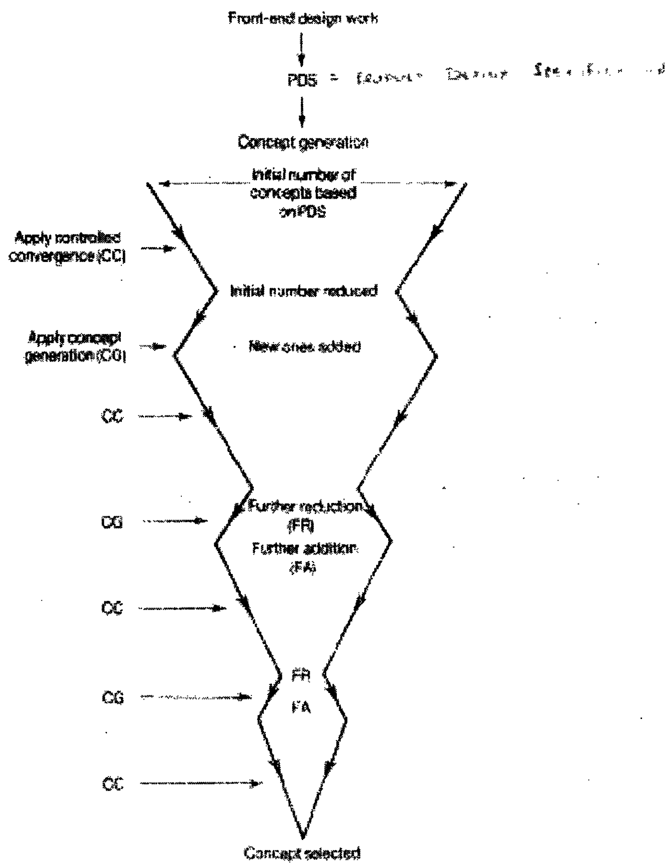


Figure 7.3: Pugh Concept Generation and Controlled Convergence [Source: Pugh, 1991]

Concept / Criteria	1	2	3	4	5	6	7	8	9	10	11
A	+	−	+	−	+	−	D	−	+	+	+
B	+	S	+	S	−	−		+	−	+	−
C	−	+	−	−	S	S	A	+	S	−	−
D	−	+	+	−	S	+		S	−	−	S
E	+	−	+		S	+	T	S	+	+	+
F	−	−	S	+	+	−		+	−	+	S
Σ+	3	2	4	1	2	2	U	3	2	4	2
Σ−	3	3	1	4	1	3		1	3	2	2
ΣS	0	1	1	1	2	1	M		1	0	2

Figure 7.4: Example Decision Matrix for Concepts compared to baseline [Source: Pugh, 1991]

NA OEM New PDS requires that PAT leaders use three spreadsheets to manage their vehicle level attributes. The first spreadsheet used is the "Attribute System Requirements Document." This spreadsheet is created for each vehicle attribute and is populated with multiple alternative subsystem design actions to deliver each attribute's targets. The inputs into this spreadsheet are the Program Attribute Leadership Strategy for that attribute, benchmarking data, brand DNA (discussed in chapter 3), and program targets. This spreadsheet is responsible for documenting a comprehensive list of proposed subsystem design changes to achieve that attribute target and the effect on all other attributes.

At NA OEM, PAT engineers are responsible for having a deep understanding of their attribute. They must know which other attributes are coupled with their attribute by common subsystem design parameters and they must understand the impact that their design decisions make on other attributes. The Attribute Systems Requirements Document, discussed above, requires that the attribute engineer paired with subsystem engineers indicate which attributes might be affected by their design actions. However, currently there is no formalized system documenting these interactions. Some documentation exists to help PAT engineers. This documentation includes design rules and quality history documents. However, most of the knowledge exists only in the minds of experienced PAT engineers. One NVH PAT leader stated that NVH PAT engineers must be able to create fishbone diagrams for their vehicle attribute that depict all design parameters that affect their attribute. However, these fishbone diagrams are not a formal PAT attribute management tool and they exist mainly in the engineer's head. Figure 7.5 shows an example of a fishbone diagram. A fishbone diagram can also be referred to as a "cause and effect" diagram or an Ishikawa Diagram, named after the originator. In a fishbone diagram, the effect under study, in our case, the attribute, is represented as the head of the diagram. Stemming from main body of the diagram are the bones which represent the causes of that effect. Each main bone can have additional bones stemming from it representing a finer level of detail of the causes.

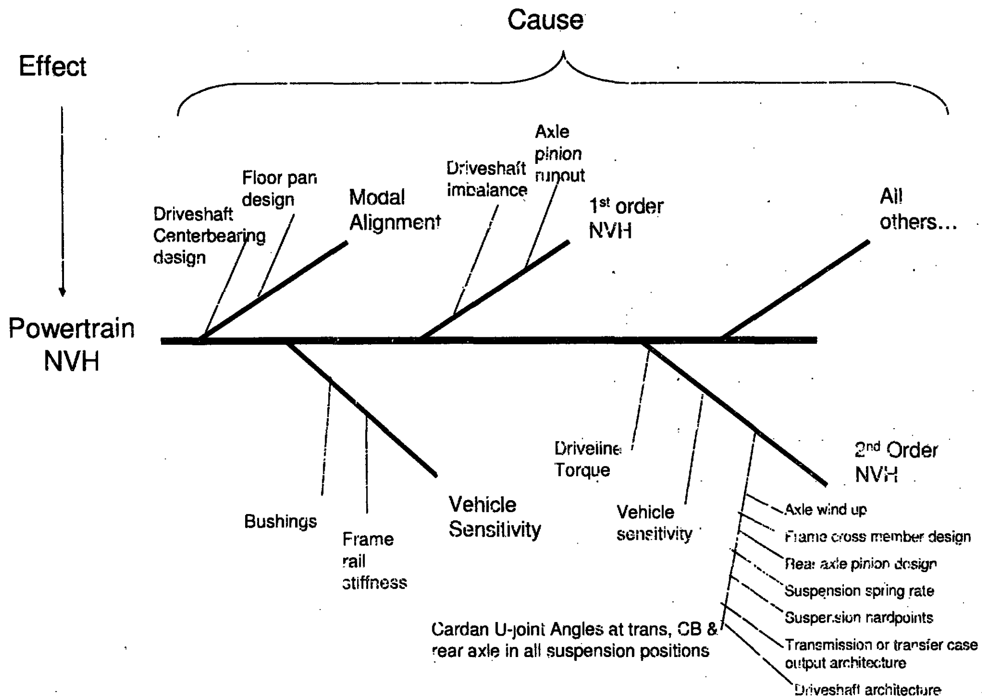


Figure 7.5: Example Causes of Powertrain NVH Fishbone diagram

The second spreadsheet used by PAT engineers is the "Initial Subsystem Design Concepts" spreadsheet. This spreadsheet is similar to the Pugh Decision matrix shown in figure 7.4 and is created for each major subsystem that is new or modified for that vehicle program. This spreadsheet documents all design concept alternatives for a single subsystem that the PAT engineers and subsequent subsystem design engineers recommend will achieve the attribute targets. It then lists these concepts against the quality, cost, weight, functional and attribute requirements and ranks each concept's performance to these requirements against a chosen baseline design.

The third spreadsheet in the attribute management process is the "Matrix of Attributes and Major Systems." This spreadsheet forces the PAT and subsystem design engineers to select the best design concepts available and present these concepts to the program for final approval. Each of the concepts is presented in this matrix and is cross-referenced with bench mark vehicles and all program quality, cost, weight, functional and attribute requirements. This matrix serves as the critical decision tool for final subsystem design concept selection. A simplified example is shown in figure 7.3.1.

Subsystem	Subsystem revision description	Program Approval Status	NA OEM Comparator	Competitive Benchmarking Vehicle	Attribute Assessment as compared to Competitive Benchmarking Vehicle
List Subsystems affected.	List all design change proposals here for each subsystem specified.	Track Program Approve/ Investigate/ Reject Status.	List NA OEM Comparator Vehicles	List Competitive Benchmarking Vehicles	Assess attribute performance as compared to same subsystem on NA OEM Comparator Vehicle

Figure 7.3.1: Simplified Sample Matrix of Attributes and Major Systems

After a subsystem concept is selected that aligns with attribute targets, the "The Functional/Attribute Plan" is created to permit assessment of functional and attribute requirements using early development tools such as CAD, CAE and rig testing.

The entire process of high level attribute trade-off, described in this section, takes place at the beginning of the product cycle plan. This process spans over a period of six months and is intended to be complete by the program target compatibility milestone.

The formalized process for attribute trade-off described above is used for high level attributes only. Section 7.2.3 of this chapter will discuss the much less formalized processes used for mid and low level attributes such as 2nd order NVH, the focus of this thesis.

If design conflicts occur between coupled attributes, these conflicts are resolved in a weekly meeting forum where all of the PAT leaders meet to ensure that the subsystem designs they are choosing to deliver their attribute targets are compatible. At this meeting design trade-offs are made via vis-à-vis discussion and negotiation. Complex analytical trade-off tools are not used.

7.2.3 Mid and Low Level Attribute Trade-Off Processes at NA OEM

New PDS does not outline formal processes for attribute prioritization and trade-off for mid and low level attributes owned at the subsystem level. Therefore, difficulty is encountered when attribute trade-offs need to be negotiated between subsystems without the leadership of a PAT. For example, as discussed in chapters four and six, 2nd order NVH, owned by the driveline subsystem team, can be improved if suspension hard points are set such that the ideal axle pinion angle is achieved at all suspension travel locations. However, high level vehicle dynamics attributes usually dominate suspension hard point decisions. Analytical trade-off tools can provide an advantage to driveline, the owner of 2nd order NVH, when making a case with the suspension engineers to allow driveline to have some influence over the suspension hard point design space. However, these tools would not remove the dominance of higher priority attributes.

It is critical for subsystems organizations, such as driveline, owning mid and low level vehicle attributes, to establish formalized communication with all subsystems that control the performance of that attribute. It is also important that the subsystems, such as driveline, that own cross-functional attributes do not assume that inputs from other subsystems cannot be adjusted to improve the attributes they own. Analytical trade-off tools can be used to assist in trade-off between two or three attributes. In the case of 2nd order NVH, analytical tools can be used to prove that vehicle dynamics performance can be maintained while 2nd order NVH is improved. Thus a more optimum solution can be found. Later in this chapter we discuss in more detail the specific systems engineering analytical tools that can be used for mid and low level attribute trade-off when dealing with only a few design parameters at one time. We will also discuss the need for subsystem engineers to understand which other subsystems influence the performance of the mid and low level vehicle attributes that they own. Currently, a single source of formal documentation of subsystem parameter effects on mid and low level attributes does not exist. This type of knowledge base may serve as a useful tool for engineers at the subsystem level owning mid and low level attributes. The 2nd Order NVH DSM discussed in chapter six was shown to be a powerful systems engineering method for capturing knowledge of subsystem parameter effects on attributes.

7.2.4 Attribute Target Decomposition and Cascading Process at NA OEM

Once high level vehicle attribute trade-offs have occurred and targets are set, the vehicle level attribute targets must be decomposed into compatible subsystem attribute targets and cascaded to these controlling subsystems. For example, 2nd Order NVH targets are a result of vehicle level NVH targets being decomposed and cascaded to powertrain and then powertrain NVH targets being decomposed and cascaded to driveline. The driveline receives many NVH attribute targets from the powertrain NVH PAT team, one of which is a target for 2nd order NVH. Attribute decomposition involves the process of resolving a vehicle level attribute target into constituent subsystem level attribute targets such that if the subsystem level targets are met, the vehicle will meet the system level attribute target once the subsystems are integrated. At NA OEM, high level vehicle system attribute target decomposition into subsystem and component level targets varies from attribute to attribute.

If the subsystem design parameters that define the attribute have a high degree of independence or non-interaction, the attribute target is decomposed hierarchically. Hierarchical decomposition of an attribute target means that a single system level attribute target can be divided up into sub-attribute targets and cascaded to subsystem such that the sum of the subsystem level sub-attribute targets equals the single vehicle level attribute target. For example, the target for the vehicle level attribute weight is decomposed and cascaded down to the subsystem level in the same way NA OEM decomposes their functional engineering organization. Weight targets are decomposed hierarchically into the subsystem level because the subsystems that define total vehicle weight are relatively independent. However, some degree of subsystem interdependence exists for weight and a trade-off exercise is performed prior to weight target cascade to subsystems. For example, the 4x4 subsystem weight can be drastically reduced by changing from a mechanical shift-on-the-fly to an electrical-shift-on-the fly. However, the powertrain controls weight will have to increase slightly to accommodate for a new controller module.

The decomposition of other attribute targets, however, such as NVH targets, is not predominantly hierarchical because the subsystems that define vehicle level NVH response are highly coupled or inter-dependent. Decisions made by many organizations combine to yield the final attribute in complex ways that are independent of the way NA OEM decomposes its functional engineering organization. The current decomposition of vehicle level NVH attribute targets to first and second levels at NA OEM is discussed in this section.

7.2.5 Vehicle NVH Attribute Decomposition and Cascade

At NA OEM, vehicle level NVH is immediately divided into the main sources that contribute to overall vehicle NVH. These main sources are discussed in section 7.2.5.1. Vehicle NVH is controlled by subsystems that are either categorized as the source, transfer path/amplifier or receiver. At NA OEM, vehicle NVH attribute targets are eventually decomposed into forcing function, magnitude, transfer path sensitivity, damping, and resonant frequency targets in the form of subsystem/component functional targets or design parameter specifications. Decomposed targets are cascaded to subsystem clusters or individual subsystems owning and controlling the source, transfer path/amplification and/or receiver. These targets are comprised of allowed maximum noise levels in decibels, frequency ranges in Hertz, and resonant frequency ranges Hertz.

7.2.5.1 First Level NVH Attribute Decomposition

At the first level of vehicle NVH decomposition, vehicle NVH is decomposed into eight major NVH sub-attributes. These eight major NVH sub-attributes are listed and described in table 7.2 Vehicle level PAT engineers are required to coordinate the highly cross-functional nature of these NVH attributes. The various NVH PAT engineers work closely with each other because these eight NVH sources are coupled and trade-offs are often needed.

NVH Sub-Attribute	Description
Road NVH	All NVH due to road surface irregularities & tire-wheel imperfections
Wind Noise	Any noise caused by air movement around the vehicle.
Powertrain NVH	All noise and vibration due to the powertrain and also including Idle NVH, acceleration and deceleration NVH, shift quality, power assisted steering and driveline forces.
Brake NVH	Brake roughness felt at the steering wheel & pedals and brake squeal.
Squeak and Rattle	Squeak is high pitched broadband transient noise caused by rubbing between two surfaces. Rattle is random transient noises causes by impacts between two surfaces.
Component Sound Quality	The level and character of the sound generated by the operation of closures, windshield wipers, adjustment motors, power locks and switches.
Pass-by-noise	The noise sent by the vehicle to the environment.
Isolation from Exterior Sound	Sound level in the interior of the vehicle caused by traffic noise, water splash and fuel slosh.

Table 7.2: Major NVH Sub-Attributes [Source: NA OEM, 2007]

7.2.5.2 Second Level NVH Attribute Decomposition

At the second level of NVH target decomposition, each of the eight major NVH attributes is decomposed into subsystem cluster level targets often named after how the NVH phenomenon is experienced by the vehicle passengers or by the major subsystem that owns the forcing function. For example, the powertrain NVH sub-attribute is broken down into many NVH attributes, some of which include "acceleration and deceleration NVH" "idle NVH," "driveline NVH" and "tip-in, tip-out clunk." At this level, NVH attribute target values are assigned to each of the NVH phenomena. These targets are derived directly from customer inputs at this level. Some of these attribute targets are in the form of objective measures such as frequency and sound level. Others are in the form of more nebulous subjective ratings based on engineering judgment. The subjective rating system used for many NVH attributes at this level pose issues that will be discussed further in this chapter.

7.2.5.3 Third and Further Level NVH Target Decomposition

At the third level of NVH attribute decomposition, NVH subsystem cluster level targets are decomposed into specific functional targets for the source, transfer path/amplifier, and receiver subsystems. Figure 7.6 shows the basic framework for NVH attribute target decomposition at NA OEM. These targets are derived from a deep understanding of the NVH phenomena. At NA OEM, subsystem and component level attribute targets are derived from several sources. These sources include lessons learned, CAE analysis, and design guides. These targets are cascaded to the appropriate engineering team by an attribute requirements database as well as with "Program Health Charts." Both of these tools will be discussed in the next section that discusses the attribute target cascade process at NA OEM. Figure 7.6 also shows that the 2nd Order NVH attribute target has not yet been formally decomposed into subsystem level targets and cascaded to the appropriate subsystems at NA OEM. This was discovered when creating the 2nd Order Attributes Management process "as-is" DSM and many of the process improvements incorporated into the "to-be" process DSM include the formalization of subsystem level attribute targets that enable vehicle 2nd order NVH targets to be met.

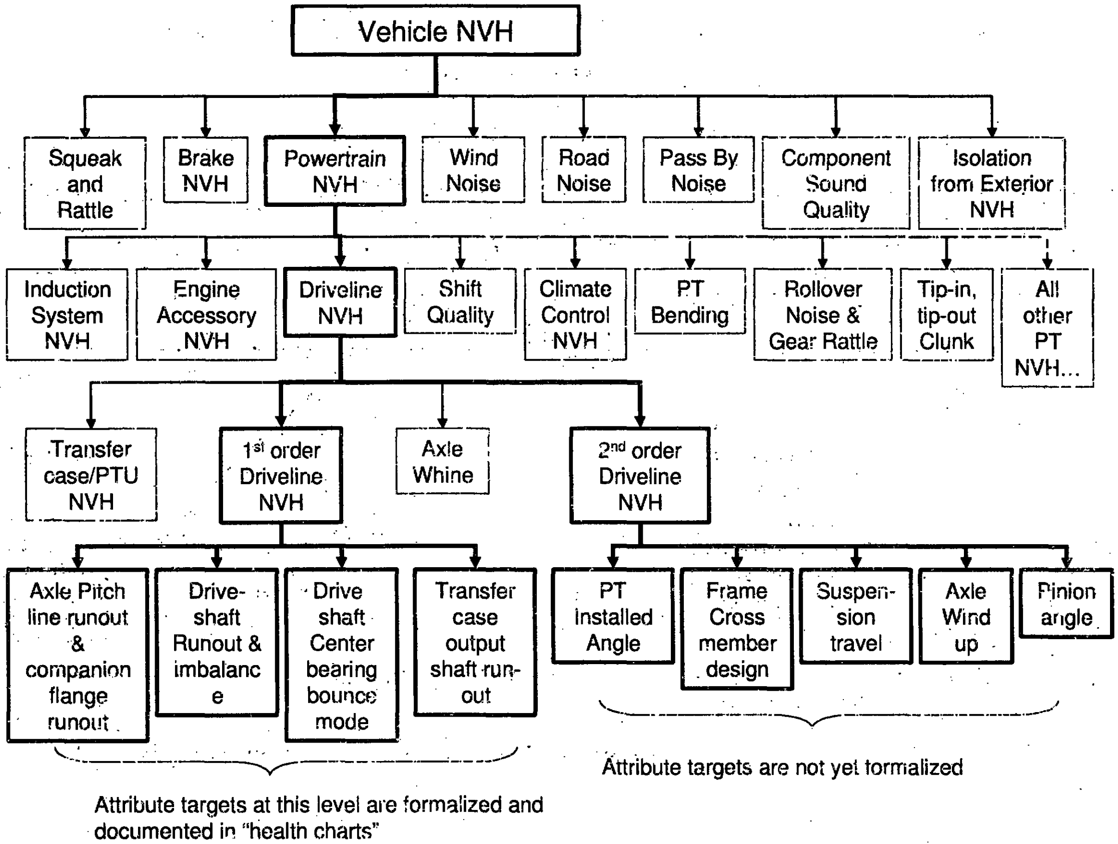


Figure 7.6: Vehicle NVH Decomposition from High Level Attribute to Component Specification

7.2.5.4 Target Cascade Process and Tools at NA OEM

NA OEM uses specific tools to cascade targets from high level vehicle attributes down to subsystem and component level functional requirements, design specifications and manufacturing capability requirements. One tool used to capture all generic attribute requirements is a single common database where reports can be pulled to find requirements down to the subsystem and component level for each of the eight major NVH types. Program specific requirements are cascaded down at each level by a cascade tool that consists of a tabulated spreadsheet that contains information on attribute target values, functional requirements, and, in some cases, design specifications. These cascade documents are referred to as "Program Health Charts" and are the core of program attribute cascade.

7.3 Current Issues with NA OEM New PDS Attribute Management Processes

After interviews with New PDS process experts, PAT leaders and subsystem engineers, it became apparent that attribute management challenges exist within the New PDS process. Engineers at NA OEM were finding issues with attribute ownership and control, trade-off and cascade. Some specific issues uncovered are discussed in this section.

One issue highlighted was the fact that attributes were allowed to be independently optimized without regard for the effect on other attributes. PAT leaders may attempt to independently optimize their own attribute at the expense of other attributes even if their attribute has been given a lower priority. This independent optimization occurs because the current culture at NA OEM rewards PAT engineers based upon the performance of their attribute alone and does not factor in necessary trade-offs. Another issue that surfaced during interviews is that there is not a clear understanding of how attributes are coupled via subsystem design parameters. This lack of engineering knowledge existed at both the system and subsystem level. System level engineers such as VE PAT engineers do not fully understand all of the subsystem design decisions that affect their attribute. At the same time, subsystem engineers do not understand the system level impacts of their designs.

Yet another issue identified is that the current culture and suboptimal processes within NA OEM PD force subsystem engineers to discount the importance of the initial design concept generation phase and instead focus their time on 1st VP, 2nd VP or launch issues for other programs. Subsystem engineers cannot expend the effort required early in a program to generate enough design concepts to enable a robust attributes trade-off exercise. Moreover, the current culture at NA OEM encourages PAT engineers to hide early vehicle level attribute problems. In this culture, PAT engineers may be forced to state to management that their targets will be met without real regard to subsystem design concept shortcomings or necessary attribute target trade-offs. This type of issue is most often seen with attribute performance vs. cost trade-offs. Often the high cost of design solutions that meet overly ambitious or inflexible attribute targets does not surface until later in the program.

Issues with target decomposition and cascade also exist. NVH attribute engineers point out that some NVH attributes targets are defined in terms of subjective ratings that can vary from engineer to engineer. The subjective rating is subject to individual engineer's opinions. These attribute targets need to be better understood and assigned objective quantitative values. The most prevalent issues in target decomposition and cascade stem from a lack of understanding of all subsystem design parameters that contribute to a specific NVH phenomenon.

7.4 A Systems Engineering Framework for Attribute Management

7.4.1. A Systems Engineering Approach to Attribute Ownership & Control

For high level attributes owned by PATs, subsystem engineers that control the performance of these attributes must become engaged in the design process early. In addition, the subsystem engineers must fully understand the implications of their design decisions on the performance of high level vehicle attributes. Often, early in the program, these subsystem engineers focus on delivering the performance requirements of their subsystem. For example, a driveshaft engineer may focus on selecting a driveshaft design concept that will likely meet subsystem level functional requirements such as maximum torque, powertrain bending, joint articulation and axial plunge requirements. The driveshaft engineer may be less likely to engage in the design concept generation exercises with the PATs required to meet high level vehicle attribute requirements. This is due in part to a lack of understanding of how their design decisions affect vehicle level attributes. Both an understanding of attribute ownership and control and a robust cascade of vehicle level attribute requirements to the subsystem level plays a critical role in ensuring subsystem engineers make design decisions based on system level attributes. In this section we will discuss how awareness of attribute ownership and control can be better promoted at NA OEM.

The systems approach recommendation after interviews with PAT and subsystem engineers is two fold. First, DSM tools can be used to assist in awareness of attribute ownership and control. Second, NA OEM PD process and culture must be deliberately modified to promote subsystem engineering involvement early in the program when design concepts are being selected to meet attribute targets.

Attribute engineers such as PAT engineers must understand the subsystems that control the vehicle level attribute they own. Subsequently, subsystem engineers must understand the impact of their design decisions on over all vehicle attribute performance, cost and quality. Here we find the DSM is a useful systems engineering knowledge capture method. A DSM can be created to map attributes as a function of subsystems. Engineers reading the DSM could immediately see which subsystems control an attribute's performance. The recommended process would be to create a generic Attribute/Subsystem DSM that displays typical dependencies between attributes and subsystems. Then, at the beginning of a new program, the generic DSM could be modified into a program specific DSM capturing unique dependencies for the specific vehicle architecture. This program specific DSM could also relay information beyond simple dependencies such as names and contact information of owning and controlling engineers and information about the type and strength of the dependency. This document should be available on the shared site for the program so that all engineers on the program can access this tool at any time. Figure 7.7 depicts the Attribute/Subsystem DSM and figure 7.8 depicts the recommended process for managing this recommended systems engineering tool.

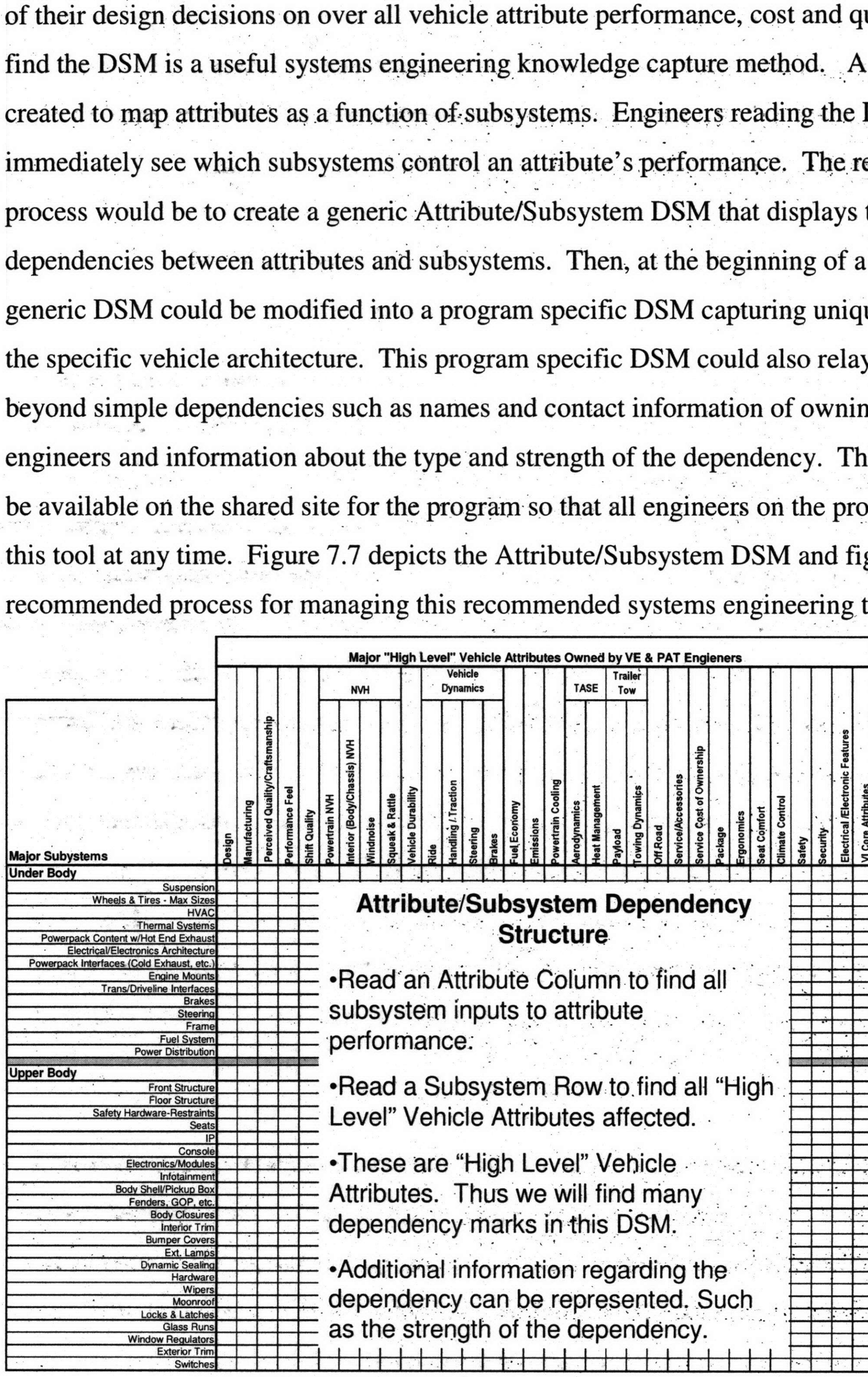


Figure 7.7: Proposed Attributes/Subsystem Dependency Structure Matrix

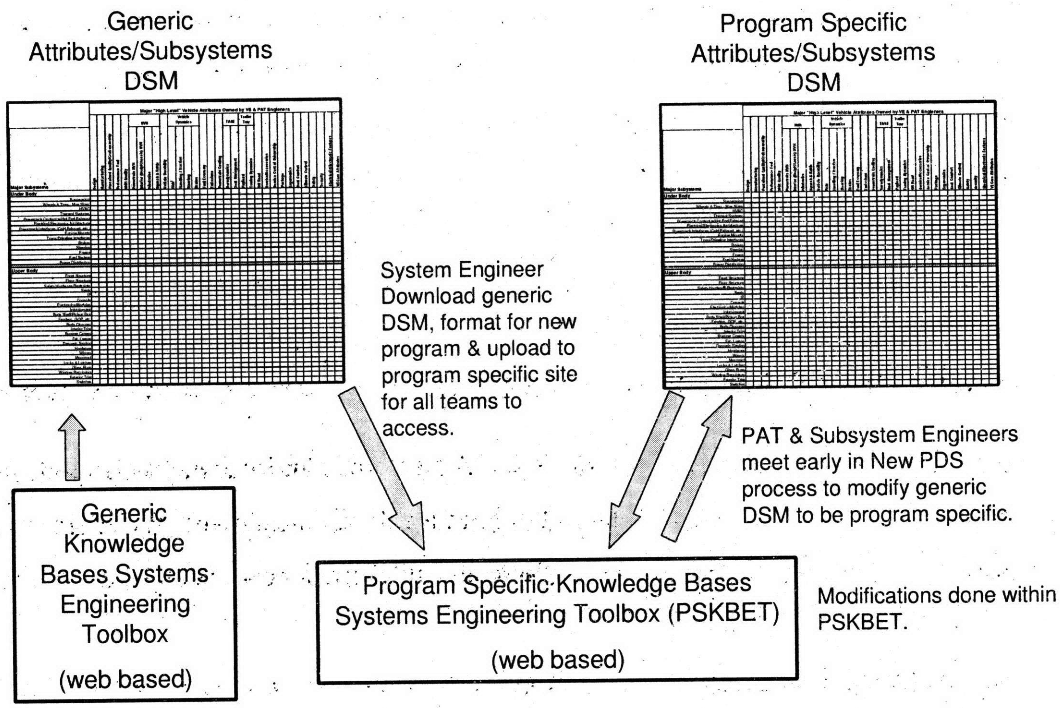


Figure 7.8: Proposed Process for Documenting PAT Attribute Ownership & Subsystem Control

Knowledge capture tools, such as the recommended Attribute/Subsystem DSM, can be created and maintained. However, these tools will not impact the quality of the process or product unless process improvements coupled with cultural changes occur at NA OEM. When questioned about why attribute management issues are still occurring with New PDS, the majority of the PAT and VE engineers responses indicated a cultural issue. The system level PAT and VE engineers indicated a lack of response from subsystem engineers early in the program during the critical stages of design concept generation. Subsequently, when questioned about attribute management issues, the majority of the subsystem engineers' responses indicated workload issues. Most subsystem engineers are working on three or more simultaneous programs and thus most of their attention is diverted first to the program that is about to launch and next to the programs at the 1st or 2nd VP build and test phase. Consequently, programs in the early stages often get "put on the back burner" because there are more pressing programs to work on with many "fires to fight."

As PD resources continue to be reduced due to corporate competitive strategies, single engineers will continue to work on multiple programs. Thus, PD process improvement becomes critical for subsystem engineering success. Methods like the task-based DSM for the attributes management process analysis, highlighted throughout this thesis, must be used to study the process, identify waste, and reduce total task time and rework. These process improvements will enable system level engineers to increase engineering efforts at the beginning of New PDS process.

7.4.2 A Systems Engineering Approach to Attribute Trade-Off Decisions

In chapter three, section three, we briefly introduced the concept of attribute prioritization. Attributes should be prioritized in order for subsystem engineers and decision makers to perform attribute trade-off studies and choose appropriate design solutions. The vector of design parameters that defines a specific design solution has associated resulting attribute values. Two or more of these attributes may depend on common design parameters. So, the question exists, which attribute should dominate the values of the shared design parameters? A simple example of this conflict between attributes is seen between 2nd order NVH and vehicle dynamics attributes. 2nd order NVH is dependent on spring rate, as spring rate affects axle pinion wind-up under high torque applications. A higher spring rate improves 2nd order NVH because there is less axle pinion wind-up under high torque applications. However, the suspension engineering team never bases their spring rate decisions on 2nd order NVH effect. In this case, vehicle dynamics attributes take precedence over 2nd order NVH when it comes to defining spring rate. However, this simplistic method of simple attribute prioritization often artificially limits the design space for many attributes. A design solution often exists that maintains acceptable values of the dominant attribute, yet improves other vehicle level attributes at the same time. In section 7.4.2.2, we will see how two or more attribute responses can be plotted vs. various design solutions and a Pareto Frontier can be developed where any solution not on the Pareto Frontier will be less desirable. In this case, multi-attribute trade-off analysis can offer value because only a few attributes are being studied and the transfer function that defines the attribute performance in terms of design parameters can be defined. Therefore, this type of analytical attribute trade-off analysis would be feasible as opposed to the more complex attribute trade-off analyses that were previously rejected by systems engineers at NA OEM.

In chapter six, the proposed "to-be" 2nd Order NVH process included new concurrent engineering between driveline and suspension engineering early in the program. The purpose of the early concurrent engineering between driveline and suspension was to explore the available design space for suspension travel and then make trade-offs between ride and handling and 2nd order NVH. These early meetings allow the driveline team to provide input into suspension travel which would enable 2nd order NVH attribute targets to be met. Previously, the driveline team assumed that the suspension travel was a fixed input. In the new process, the driveline team has input into the suspension travel early in the design process to achieve improved 2nd order NVH response.

In order to perform attribute trade-off exercises, engineers must understand 1) the design parameters that affect each of the attributes in the trade-off study, 2) how the attributes are coupled by design parameters, and 3) the function that defines the attribute response in terms of design parameters. The last item, "understanding the transfer function that defines attribute response in terms of design parameters" has proven to be the most difficult. Section of 7.4.4 of this chapter will discuss a suggested attribute knowledge base tool that would map these relationships.

*7.4.2.1 **Attribute Focused Task Based DSM: Defining Attributes as a Function of Design Parameters***

Going back to our case study attribute focused task based DSM, we find that three types of tasks emerge, 1) Design Space Negotiations, 2) Design Parameter Setting, 3) Verification. The attribute focused task based DSM creation process enabled us to discover and document all of the cross-functional design parameters that affect 2nd order NVH. Due to the fact that this DSM focused on only a single attribute we do not see the full picture on attribute coupling. However, the DSM creates a useful base for 1) Completely understanding all design parameters that describe 2nd order NVH 2) knowing which design parameters are currently assumed as input only and which have feedback loops and 3) knowing which subsystem design parameters may have conflicting objectives due to higher priority attribute targets. Therefore, the attribute focused task-based DSM serves not only as a process improvement method, but as an excellent tool for attribute management improvements. Users of the DSM can question why certain subsystem design parameters are accepted as input only to attribute performance and then ask if there is any available design space for attribute performance improvement by changing these input-only parameters. Users of the DSM can also fully define the design parameter variables that define the attribute. Thus, this serves as a base for constructing the parametric functional equation describing the attribute in terms of design parameters.

7.4.2.2 Multi-Attribute Trade-Off Analysis

Although vehicle level engineers at NA OEM discourage the use of technical multi-attribute trade-off analysis (MATA) tools for high-level attribute trade-off amongst many attributes controlled by hundreds of design parameters, MATA can be useful when performing attribute trade-off analysis at the subsystem level for mid and low level attributes. MATA provides systems engineers with an analytical tool to identify Pareto Superior options for finalizing a system design with conflicting attributes. [Tabors & Hornby, 2005] This type of analysis is most useful when making trade-offs between two or three attributes that share a relatively small vector of design parameters. This is because the Pareto Frontier of solution sets can be easily visualized as a frontier curve or as a surface for two or three attributes respectively. The utility of MATA is that it helps systems engineers explore large design spaces. Even when the total number of decision variables is deliberately kept low, the ranges of values that each decision variable can take on is large and thus the design space can grow exponentially. MATA helps tackle this large design space. [El-Rayes & Kandil, 2005] For example, MATA can be used to set a vector of common frame, suspension and driveline design parameters such that targets for dominant attributes such as ride quality are met and at the same time 2nd order NVH is optimized within the available design space. Many algorithms exist to vary a vector of design parameter decision variables such that two or three objective functions are optimized to find a Pareto Optimal Frontier (two attributes) or Surface (three attributes). This thesis will not go into the details of the algorithms that identify the Pareto Optimal Frontier or surface. Instead, the main concepts behind MATA and its application to mid and low level vehicle attribute trade-off are discussed next in this section.

Once decision variables (in this case these are design parameters), objective functions (in this case minimizing or maximizing attribute performance), and constraints are defined, algorithms can be used to identify a Pareto Frontier or Surface of design solutions. A Pareto Optimal frontier or surface is defined by decision variable solution sets that maximize or minimize the objective functions such that if the value of any of the design parameters is changed and the solution is not on the frontier, then the solution is inferior, with one or more of the attributes' performances being degraded. Thus, all solutions along the frontier or on the surface are considered Pareto Optimal and all should be considered when choosing the final set of decision variable values. Figure 7.9 is a simple depiction of how the design space for design parameters is translated into a performance space by performance objective functions for attributes F1 and F2.. The objective function in this case is to minimize F1 and F2. The white arrow represents the transfer function that defines attribute performance in terms of subsystem design parameters. The Pareto Frontier is created by the design parameter solution sets, in this case values for X1 and X2, that provides an optimized solution for both objective functions where any movement within the design space away from the Pareto Frontier will degrade one or both of the attributes' performance. In figure 7.9 we see the Pareto Frontier of the solution space highlighted. As we move along the Pareto Frontier attribute performance is traded off between attributes F1 and F2. Thus, it is up to the design team to decide if one attribute takes priority over another. In the example of figure 7.9, the utopia point would be the (0,0) point for F1 and F2. However, there is no combination of X1 and X2 that will achieve this point. Instead we have to stay on or behind the frontier. But there is one or possibly several places on the frontier that are close to this utopia point and it is here that we should seek solutions.

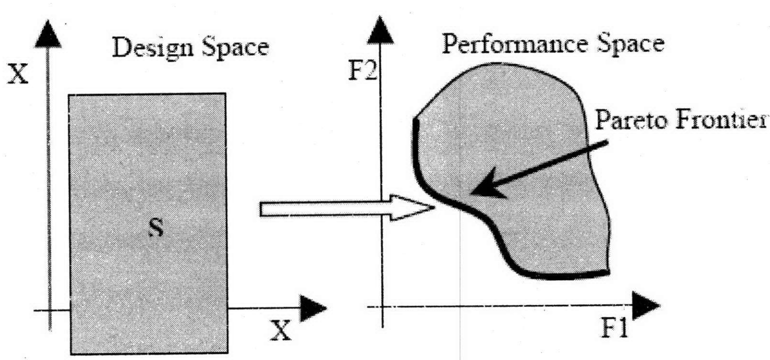


Figure 7.9: Mapping from Design Space to Performance Space with Objective to Minimize F1 and Minimize F2 [Source: Agrawal et al., 2004]

In order to assure that high priority attribute targets are met, the engineering team performing the MATA must allow the higher priority attribute to set bounds on the design space for acceptable performance. These bounds on acceptable performance, as opposed to a specific value for acceptable performance, enable the design parameter values to be adjusted, within the bounds, until other attributes are improved. For example, rather than just assuming that ride quality controls spring stiffness, one can search for a value of spring stiffness that allows ride quality targets to be met and reduces the 2^{nd} order NVH attribute response at the same time.

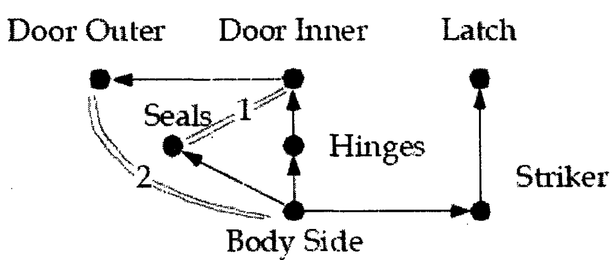
7.4.3. A Systems Engineering Approach to Attribute Decomposition and Cascade

In order for system level attribute targets to be decomposed into subsequent subsystem level attribute targets, the subsystem level design parameters and interfaces controlling the system level attribute response must be well understood. Knowledge based tools for documenting subsystem interactions and design parameter effect on system level attribute performance will be discussed in the next section. This section will focus on suggested systems engineering methods for decomposing system level attribute targets into compatible subsystem level attribute targets and further decomposing these subsystem level attribute targets into component level attribute targets. It is the responsibility of the vehicle level systems engineers owning the attribute to understand how to effectively decompose their attribute target into various compatible subsystem attribute targets. It is critical for the subsystem level attribute targets to be compatible and aligned with the program's attribute, quality, cost, weight, and functional targets. An attribute engineer may force overly ambitious targets on all of its contributing subsystems, unnecessarily driving higher product costs and engineering resources. These overly ambitious targets are not compatible with program cost targets. It is important to note that it is not the sole responsibility for the vehicle level engineers to set and cascade attribute targets. These engineers do not have a full understanding of subsystem function. Thus, vehicle level engineers must work closely with subsystem engineers when developing and setting attribute targets.

As discussed in this chapter, specific methods for attribute decomposition vary widely based on the attribute and the controlling subsystems. Thus, each attribute has to be decomposed on a case by case situation. Only a high level frame work for decomposition with suggested methods and tools would be of help to a systems engineer tasked with decomposing a vehicle level target into subsystem level targets. Systems engineering methods and tools exist to assist in the process of vehicle level attribute target decomposition. A four step framework for attribute decomposition and cascade along with suggested systems engineering methods and tools at each step are presented below and shown in figure 7.11.

The first step in decomposing an attribute target is to identify the subsystems that are responsible for controlling that vehicle level attribute. It is these subsystems that will have attribute targets cascaded to them. Four systems engineering methods that can assist in identifying the controlling subsystems are Datum Flow Chains, Design of Experiments, Ishikawa Fishbone Diagram paired with the "Five Whys Method," and the DSM. These four systems engineering methods are summarized below:

Datum Flow Chains: In a Datum Flow Chain (DFC) directional arrows represent a link between two parts. The direction of the arrow identifies one part as having the responsibility to locate the other part and defines constraints. There are six total degrees of freedom represented in a DFC which includes x,y,z directions and three degrees of rotation. Figure 7.10 represents an example DFC for a door. The red lines represent Key Characteristics of the door and are equivalent to attributes like fit and finish or closing effort. [Whitney 2004, Noor 2007]



1 = Part relationship that defines "Door Closing Effort" Attribute

2 = Part relationship that defines "Fit and Finish" Attribute

Figure 7.10: Example DFC [Source: Whitney, 2004, Noor, 2007]

In "Mechanical Assemblies: Their Design, Manufacturing and Role in Product Development," Daniel Whitney instructs reader on the details of the use of Datum Flow Chains to represent mechanical assemblies. Jehanzeb Noor's 2007 MIT Thesis supports the claim that DFC is a useful attributes management tool. This thesis asserts that DFC could be a potentially useful tool for attribute decomposition. This is a powerful tool for finding the subsystems and components that affect an attribute's response.

Design of Experiments: Design parameter effect on attribute performance can be discovered from a design of experiments. This thesis will not go into the details of how to conduct a Design of Experiment. If the reader would like to learn how to conduct and analyze a design of Experiments the existing literature is abundant.

Ishikawa Fishbone Diagram and "Five Whys Method": Refer to detailed discussion in section 7.2.2. This is another useful tool for identifying attribute dependencies on subsystem design parameters.

Attribute-Dependency Structure Matrix (also referred to as DSM): Refer to detailed discussion in section 7.4.1. This tool can play a critical role in documenting lessons learned from previous programs and communicating subsystem and attribute information between the vehicle system and functional subsystem engineers. A vehicle engineer tasked with decomposing a vehicle level attribute target can use this matrix to determine which subsystems affect the performance of that vehicle level attribute target based on knowledge gained from previous vehicle programs.

The second step in the prescribed framework for decomposing an attribute target is to determine a balanced attribute target value for the controlling subsystems. A balanced attribute target is defined as a target that is deemed attainable with current available technology and resources and is deemed compatible with all other cost, functional and attribute targets. Systems engineers must look to robust tools to assist in setting balanced target values that can be cascaded to the appropriate subsystems. The best tool for setting target values early in the program is valid vehicle CAE models. In the case of NVH, the vehicle CAE model should be used to determine predicted vehicle sensitivity to NVH forcing function inputs. The transfer function must be well understood and accurately represented in the model. Subsystem and component resonant frequencies must be well understood so that appropriate modal alignment avoidance strategies can be cascaded. It is also critical that subsystem/component manufacturing capabilities are included in the model. Manufacturing variation can drastically change the NVH response of a system. Monte Carlo analysis can be performed to ensure acceptable vehicle NVH response within the manufacturing capabilities. For example, 1st order NVH targets are cascaded from the vehicle level to driveline. Driveline further decomposes this 1st order NVH target into targets for the controlling subsystems which are transmission/transfer case, axle and driveshaft. Careful CAE and Monte Carlo analysis is performed to make sure that balanced 1st order NVH targets are cascaded to each of these subsystems. If the manufacturing capability for one of these subsystems only allows a minimum imbalance of 0.4in-oz, then it would not make sense to require more stringent imbalance targets for the other systems. This is because the imbalance of one subsystem such as rear axle can be offset by the imbalance of another subsystem such as driveshaft. Thus, the goal is to match the imbalance level such that the subsystem imbalances cancel each other out at the planes of interface.

The third step in the prescribed attributes decomposition and cascade framework is to cascade the balanced targets to the appropriate subsystems and components. Cascade tools such as those used by NA OEM are effective. As discussed NA OEM uses a standard spreadsheet tool for all attribute cascades. This tool enables a clear expectation and understanding of attribute targets and a forum for attribute target negotiation between the system and subsystem engineers. The final step is to document all lessons learned to build an attributes knowledge base. A systems engineering approach to creating an attributes knowledge base is discussed in the next section.

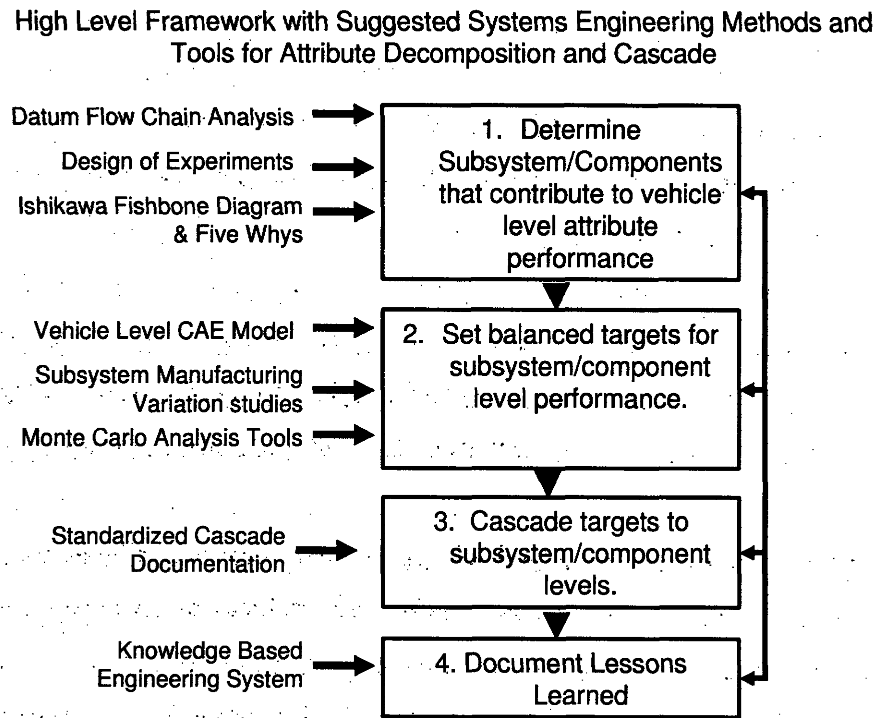


Figure 7.11: High Level Framework with Suggested Systems Engineering Methods and Tools for Attribute Decomposition and Cascade

7.4.4 A Systems Engineering Approach to Attribute Knowledge Base Management

[NVH Technical Expert]'s team worked for most of one week before they isolated the culprit -- a component in the drivetrain. Seeking greater durability, engineers had changed the material without consulting on the NVH implications. [NVH Technical Expert] was able to recommend an alternative design specification that could be put in place immediately; the fix was made, and the line rolled on.

-NA OEM

The above quote is one example of the effect of a poor understanding of subsystem level design effects on vehicle system level attributes on product quality. A lack of tacit engineering knowledge and/or access to documented engineering knowledge is the root cause of this vehicle production issue. This situation also highlights a lack of communication between the subsystem and system level engineers. In this situation, a design change was made at the component level without an understanding of the effect on vehicle level attributes and without communication to vehicle system level engineers. The question exists now, can an engineering knowledge based tool help prevent situations like that quoted above?

An engineering knowledge based tool that documents and displays subsystem level design effects on attribute performance could serve to facilitate an understanding and communication between system level vehicle engineers and subsystem functional design engineers. This knowledge based tool could document the relationship between attribute performance and subsystem design parameters. Thus, if a subsystem/component is being changed, the functional design engineer will be able to assess what attributes are affected. Moreover, a system level vehicle engineer would have a high level view of all subsystem design parameters that affect the performance of the attributes that they own. As mentioned in chapter three, attributes are coupled by common subsystem design parameter dependencies. Once again, we present a representation of attribute coupling by subsystem design parameter in figure 7.12. The DSM could serve as a useful tool to store and depict attribute dependencies on subsystem design parameters. However, the more comprehensive this DSM becomes the more difficult it would become to read. An alternative method for depicting parameter-attribute relationships is an attribute/Subsystem/Design Parameter three dimensional visualization tool. This tool could add value to a Product Development organization. The concept behind such a visualization tool is shown in figure 7.12. This concept would need to be further developed and a software program created to help show parameter-attribute relationships and attribute coupling.

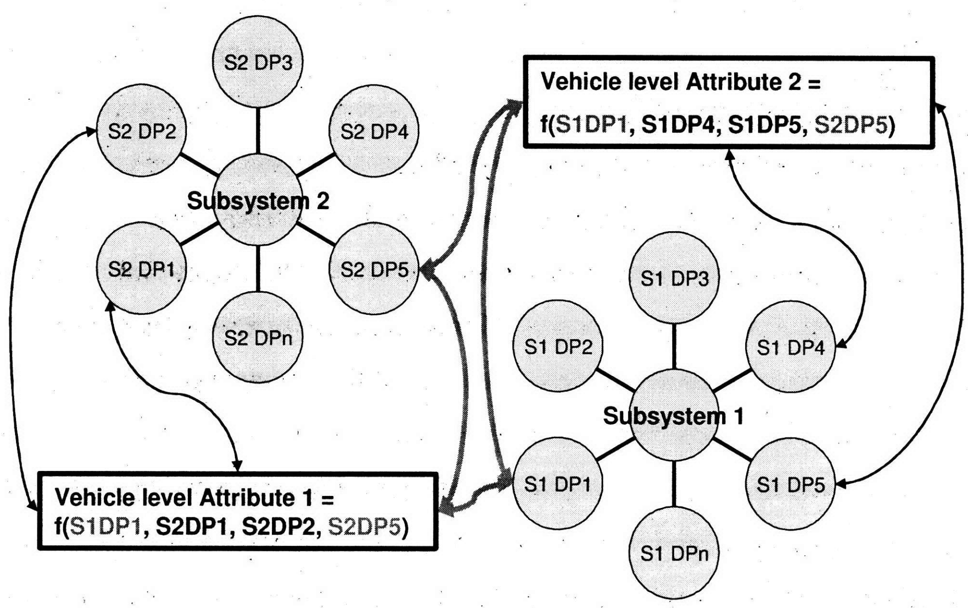


Figure 7.12: Attribute/Subsystem Dependency and Attribute Coupling Representation

7.5 Chapter Summary

In this chapter we defined high, mid and low level vehicle attributes depending on the total number of subsystems that contribute to the performance of an attribute. The research at NA OEM revealed that formalized attribute management processes exist for high level vehicle attributes. However, the process of attribute management for mid and low level attributes is not formalized at NA OEM. This conclusion is consistent with our finding for the mid-level vehicle attribute known as 2nd order NVH discussed throughout this thesis. Current attribute ownership and control as well as trade-off methods at NA OEM are discussed. The research found that for high level vehicle attributes, complex technical trade-off analysis tools have been unsuccessful. Currently, NA OEM relies on an attribute trade-off and subsystem concept selection process that mirrors the Pugh concept convergence method and Pugh Decision Matrix. This process relies heavily on engineering face to face meetings to complete target trade-off. Trade-off processes are not formalized for mid and low level attributes owned at the subsystem level such as 2nd order NVH. Thus, it is critical for subsystems owning mid and low level vehicle attributes to establish formalized communication with all subsystems that control the performance of that attribute. It is also important that these subsystems, owning cross-functional attributes, do not assume that inputs from other subsystems cannot be adjusted to improve the performance of the attribute they own.

Current attribute decomposition and cascade processes at NA OEM are explored. The decomposition of highly coupled attributes such as powertrain NVH is described. This research finds that the current processes used at NA OEM can be paired with systems engineering principles methods and tools to arrive at a basic framework for attribute decomposition. In particular, DSMs like those developed in chapter 6 for 2^{nd} order NVH are useful to document these interactions. As discussed earlier, this chapter presents a high level overview of what is a potentially rich topic of research. Chapter 8 of this thesis will discuss this as a useful area of future research.

8. CONCLUSIONS AND RECOMMENDATIONS

This chapter presents conclusions and recommendations derived from this research where the processes of vehicle level attribute management at a North American Automotive OEM were studied in detail. Specifically, the research studied the current processes and the product development organizations that deliver a vehicle system design that meets 2nd order NVH attribute targets. The Design Structure Matrix method for process modeling and restructuring was applied to the current processes used to deliver a vehicle level design that meets 2nd order NVH attribute targets. This attribute is highly cross-functional in nature, yet is owned at the subsystem level by the driveline systems team. The DSM method was able to successfully enable the process improvement team to restructure the current process such that engineering lead time and resources are reduced and product quality is increased.

This research also examined new attribute management processes recently implemented by the introduction of a new Product Development System. Attribute trade-off, decomposition and cascade processes prescribed by the new Product Development System are examined. Systems engineering principles, methods and tools are suggested for attribute management improvement. Systems engineering principles, methods and tools were suggested for attribute trade-off, decomposition and cascade. The topic of attribute trade-off, decomposition and cascade principles, methods and tools begs for future research. A comprehensive study of attributes management at other enterprises would bring value to this topic.

8.1 Conclusions

8.1.1 DSM Method for Process Modeling and Restructuring Conclusions

The team responsible for creating the DSM model found the DSM creation process itself to be of value to the organization. The process of mapping the current 2nd order NVH attribute management process led to the emergence of lessons learned from previous programs, notable process issues, and process improvement ideas. Most of the process restructuring ideas surfaced during the DSM creation process. Thus, we conclude that the mere process of creating a DSM plays a large role in generating process improvement strategies. This finding supports the research of Cronemyr et al which states, "One could argue that the main process improvements come from the knowledge gained from the process mapping itself, i.e. once the map exists, the suggestions for improvements become obvious" [Cronemyr et al., 2001]. All Product development organizations involved in the creation process also found the task list generated prior to the DSM process mapping exercise to be of value. The task list enabled the team to identify concrete tasks and associated deliverables. In some cases, the task list identified missing deliverables, and in that case a standardized deliverable was created.

The DSM was proven to be a useful tool to demonstrate the effects of process changes on process lead time and variance to upper management when these process changes require resources to implement. Again, this supports the research of Cronemyr et al. which asserts that the DSM Simulated To-be/As-is Ratio, STAR, can be used to assess the impact of process restructuring [Cronemyr et al., 2001]. However, the research in this thesis concludes that DSM model and associated input data used to generate simulation results used to justify investments in process restructuring should be verified for validity by an unbiased team. Any time an analytical model is used to make decisions, biased creators of the model can "play the system" and easily skew the results to indicate any magnitude of improvement desired. This is especially true in the case of DSM model simulations. In order to simulate the proposed "to-be" process, estimates must be made on the decrease in rework probabilities, and task time and increases in the learning curve and rework impact. Users of the model can easily artificially inflate these estimates in order to achieve the desired simulation results. Thus, it is critical that users of the model are unbiased and that the input data to the model is verified to be accurate by the entire team.

The driveline systems engineering team, owning the 2nd order NVH vehicle attribute, found the Task-based DSM process model created to serve as a useful knowledge base tool. The completed "as-is" DSM and "to-be" DSM both serve as a useful reference tool for engineers. Engineers are able to quickly discover which tasks they own and what deliverables they need to complete each task. Engineers are also able to identify who needs the information they create. Prior to the creation of the DSM, no such documentation existed. This tool also enables engineers to visualize the iterative nature of the process of 2nd order NVH attribute management. The DSM enables the driveline systems engineers to get a holistic view of the complex and iterative process where flow charts become too complicated to read [Cronemyr et al., 2001]

The DSM model created in this thesis does not take into account vehicle program complexity.. Programs of varying complexity will have varying process lead time and variance. Therefore, this thesis concludes that historical task time and iteration data is needed from six or more vehicle programs of similar complexity to make the data for a model statistically significant. Then, these data sets must be collected from programs of varying complexity such that vehicle program complexity can be correlated to process lead time. Then, a scaled model of the process that takes into account the complexity differences could be created to more accurately estimate process lead time for all programs.

8.1.2 Management of Cross-Functional Attributes Owned at Subsystem Level Conclusions

This research at NA OEM revealed that the "mid level" vehicle attribute known as 2nd order NVH has historically caused a high incidence of vehicle launch issues and warranty claims. This vehicle level attribute is not owned by vehicle engineering. Rather, this vehicle level attribute is owned by the driveline subsystem. Thus, the driveline subsystem engineers are tasked with ensuring that the subsystems that contribute to 2nd order NVH performance are designed and tolerances controlled such that the vehicle meets customer defined requirements for all vehicle produced. The task of coordinating the design parameters of subsystems outside of the driveline organization has proven to be challenging. Currently other subsystems that control 2nd order NVH vehicle performance reside within other PD organizations with management chains that only merge with driveline's management chain at the vice president of North American Product Development. Additionally, there is no formalized process of communication between driveline engineers and the other subsystem engineers owning design parameters that affect this attribute. The New PDS at NA OEM does not provide process governance of mid-level vehicle attributes such at 2nd order NVH. Thus, subsystems owning these attributes, such as driveline, are left at the subsystem level to derive their own local processes. This research mapped the current processes used within the driveline organization to manage the development and integration of subsystem designs such that 2nd order NVH targets are met. The use of systems engineering methods created an improved process for managing this highly cross-functional vehicle attribute, owned at the subsystem level. Key lessons were learned for the improvement of managing cross-functional vehicle attributes such as 2nd order NVH. These are summarized in this section.

The DSM creation process revealed a disconnect between the deliverables created and the deliverables needed by various cross functional teams when engineering their subsystems to enable 2nd order NVH vehicle level targets to be met. This finding supports Antoine Guivarche's assertion in his 2003 thesis [Guivarch 2003]. The DSM creation process enabled cross functional teams to come to a consensus on what specific information is required, what format this information should be presented, and where this information should be stored. Additionally, this research found that the timing of information creation is equally as critical as having the right information. In essence, the right deliverable is needed at the right time. The DSM highlighted critical input tasks, a single task that feeds many other tasks, and information bottleneck tasks, a single task that requires information from many other tasks. These tasks depend on timely creation of information. Thus the team focused on the timing of the deliverables created by these tasks.

Vehicle level attributes owned at the subsystem level must be managed by formalized cross-functional communication. This research found that there was no formalized cross functional communication in the management of 2nd order NVH. In the "as-is" process, "Good" driveline systems engineers knew that they had to ask for specific information from various subsystems. The new "to-be" process formalizes cross-functional communication in the form of face to face meeting at strategic times during the development phase.

This research found that driveline engineers owning 2nd order NVH vehicle attribute performance assumed some design parameters affecting 2nd order NVH as pure inputs. Inputs such as axle pinion travel patterns during suspension movement were assumed to be one way input with no feedback loop between driveline and suspension engineers. The driveline engineers said that axle pinion movement patterns due to suspension geometry played the largest factor in determining the 2nd order NVH vehicle response. However, no driveline engineer attempted to influence suspension geometry such that the ideal axle pinion movement was achieved. The DSM creation process allowed the driveline engineers to discover that they could become involved in the suspension geometry design early in the program to achieve more desirable axle pinion movement. Thus, the lesson learned is that engineers owning vehicle attribute performance should not assume design parameters as pure inputs. Feedback loops between the engineer owning the attribute performance and the engineering owning the subsystem design should be established.

The DSM method for process modeling proved to be a valuable tool for improving the total time to develop, integrate and test subsystem design to meet vehicle level attributes that are affected by many different subsystems. The application of the DSM methods for process modeling and improvement was able to create a predicted 30% reduction in average engineering time dedicated to meeting 2nd order NVH attribute performance targets and a 45% reduction in lead time variation. However, the DSM creation process is time consuming and required expertise in the DSM method itself as well as in the process under study. Therefore, the DSM method for process modeling is recommended only for those vehicle level attributes deemed as "high impact." Again "high impact" attributes are defined as attributes that have caused significant launch issues in the past or are causing high incidence of warranty claims.

8.1.3 2nd order NVH Management Specific Conclusions

The DSM process modeling method yielded several critical process improvements in ensuring that a vehicle meets 2nd order NVH targets. In addition, these process improvements were predicted to improve the total process lead time and variation and reduce objectionable 2nd order NVH issues occurring at vehicle launch or in the field. The significant 2nd order NVH process improvements are summarized below:

- The creation of formalized cross-functional meetings early in the program cycle to establish early assumptions and understand design space.
- The creation of a database of early information and assumptions to pull ahead initial subsystem design.
- The improvement of current desk top analytical tools to expedite early design iteration.
- The establishment of well defined objective 2nd order NVH targets based on lessons learned from previous similar programs and on early program information.
- The adaptation of current driveline rig to 2nd order NVH testing capabilities to reduce the probability of rework after 1st vehicle prototype testing.

8.1.4 Attribute Ownership and Control, Trade-off, Decomposition and Cascade Conclusions

Ownership and Control: Research on the topic of powertrain NVH attribute management at NA OEM found that vehicle level attributes that are affected by most major vehicle subsystem are owned and managed by vehicle engineers on Program Attribute Teams. However, some vehicle level attributes are owned at the subsystem level and subsystem engineers owning a vehicle level attribute that is influenced by many other subsystems are responsible for coordinating the design efforts. The use of an Attribute/Subsystem DSM may help both vehicle level engineers owning attributes and subsystem engineers owning vehicle level attributes better understand who controls design parameters that influence the performance of that attribute. Moreover, an Attribute/Subsystem DSM may help a subsystem engineer focused on the design of their system understand who owns the attributes that their design affects.

Attribute trade off: The research at NA OEM found that for high level attribute trade-off, where subsystem concepts are not yet defined and potentially hundreds of subsystem design parameters need to be traded off, technical multi-attribute trade-off analysis (MATA) failed. Instead, NA OEM now relies on an attribute trade-off process that occurs right at the beginning of a new vehicle program and closely parallels the Pugh Concept Selection Process. Here attributes are traded off as subsystem design concepts are proposed. At the end of the process subsystem design concepts are chosen based on their compatibility with all attribute targets. However, the success of this method depends critically on involvement of subsystem engineers whose time is presently taken up by work on programs that are closer to launch.

MATA is recommended for small attribute trade-off analysis for mid and low level attributes. The recommendation is to use MATA to trade off performance between two or three attributes with only a few controlling design parameters. This research identified two significant disadvantages to MATA First, it is usually very difficult to accurately derive the transfer function between design parameters and attribute performance. Second, anytime analytical tools are being used to make decisions the team must be aware of potential gaming and biasing of the input data. Analytical tools cannot be treated like black boxes turning out answers. The inputs must be carefully studied. The transfer functions must be carefully understood and continually updated as more information is learned.

Decomposition and cascade: This research found that attribute decomposition methods vary widely depending on the attributes. Therefore, a basic framework for attribute decomposition with suggested systems engineering methods and tools is recommended. This framework paired with the recommended systems engineering methods and tools can be used to guide engineers owning vehicle level attributes to decompose a vehicle level attribute target into compatible subsystem level targets. NA OEM currently uses a standard cascade tool for all vehicle level attributes owned by vehicle engineering Program Attribute Teams. This tool consists of a MS Excel workbook of spreadsheets that clearly defines the breakdown of an attribute target at one level to the next level down. This research found that attribute cascade is not one directional. At NA OEM attributes targets are initiated at the vehicle level but are developed and negotiated by teams of vehicle level and subsystem engineers. The decomposition and cascade of vehicle level attributes is a recommended topic for future research this will be further discussed in section 8.2.

8.2 Recommendations for Future Research

Further research is needed in the area of attributes decomposition and cascade. Specifically, research should be done across multiple OEMs or industries to search for patterns in attribute decomposition to establish a more detailed and useful framework for attribute decomposition. Additionally, the industry can be searched for effective target cascade tools and methods.

Other potentially valuable areas of future research include the following: 1) Investigating the impact on attribute performance of synchronizing deliverable timing at NA OEM and measuring the effect of deliverable delays on total process lead time and product quality. 2) Creating a DSM process model that can be scaled to represent program complexity.

REFERENCES

Adams, Douglas P. "Universal joint", in AccessScience@McGraw-Hill, http://www.accessscience.com.libproxy.mit.edu, DOI 10.1036/1097-8542.721800

Agrawal, G et al. "Intuitive Visualization of Pareto Frontier for Multi-Objective Optimization in n-Dimensional Performance Space." Collection of Technical Papers – 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference. 2 (2006): 729-742.

Browning, Tyson R. "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions." IEEE Transactions on Engineering Management 48.3 (2001) 292-307.

Cho, Soo-Haeng, "An Integrated Method for Managing Complex Engineering Projects Using the design Structure Matrix and Advanced Simulation," SM Thesis, MIT Libraries, Cambridge, MA: 2001.

Crawley, Edward. IAP Systems Architecture Lecture 2. Massachusetts Institute of Technology, Cambridge, MA. 17 January 2006.

Cronemyr, Peter et al. "A Decision Support Tool for Predicting the Impact of Development Process Improvements." Journal of Engineering Design 12.3 (2001): 177-199.

deWeck, Olivier and Reinhard Haberfellner. Systems Engineering: Principles, Methods and Tools for System Design and Management, 1st ed. Cambridge: Unpublished and Presented at Massachusetts Institute of Technology, 2006.

Dong, Qi, "Representing Information Flow and Knowledge Management in Product Design Using the Design Structure Matrix," SM Thesis, MIT Libraries, Cambridge, MA: 1999.

El-Rayes, Khaled, and Kandil, Amr. "Time-Cost-Quality Trade-Off Analysis for Highway Construction" Journal of Construction Engineering and Management. Apr. 2005: 477-486.

Eppinger, Steven D. "Model-based Approaches to Managing Concurrent Engineering." Journal of Engineering Design, 2.4 (1991): 283-290.

Guivarch, Antoine D., "Concurrent Process Mapping, Organizations, Project and Knowledge Management in Large-Scale Product Development Projects Using the Design Structure Matrix Method," SM Thesis, MIT Libraries, Cambridge, MA: 2003.

Guivarch, Antoine D., and Whitney, Daniel E. "Concurrent Process Improvement, Organizations, Project and Knowledge Management in Complex Product Development Projects Using the Design Structure Matrix (DSM)." Proceedings of DTEC '04: ASME 2004 International Design Engineering Technical Conferences. Salt Lake City, Utah. (2004).

Hauser, John R., and Clausing, Don. "The House of Quality." Harvard Business Review. 66.3 (1988): 63-73.

Krishnan, Viswanathan et al. "A Model-Based Framework to Overlap Product Development Activities." Management Science 43.4 (1997): 437-451.

Larkin, John. "Ford's New Way Forward." Automotive Industries Dec. 2006: 75-77.

Loureiro, Geilson et al. "Systems Engineering Framework for Integrated Automotive Development." Systems Engineering 7.2 (2004): 153-166.

Maier, Mark W. and Eberhardt Rechtin. The Art of Systems Architecting 2nd ed. Boca Raton: CRC Press, 2002.

McGill, Eric A., "Optimizing the Closures Development Process Using the Design Structure Matrix," SM Thesis, MIT Libraries, Cambridge, MA: 2005.

Morgan, James M. and Jeffrey K. Liker. The Toyota Product Development System. New York: Productivity Press, 2006.

Noor, Jehanzeb G., "A Comprehensive Approach to Complex System Product Development: Operations Management Tools applied to Automotive Design," SM Thesis. MIT Libraries, Cambridge, MA: 2007.

Pugh, Stewart. Total Design: Integrated Methods for Successful Product Engineering. Addison-Wesley Pub. San Diego: 1991.

Smith, Robert P. and Steven D. Eppinger. "A Predictive Model of Sequential Iteration in Engineering Design." Management Science Aug. 1997: 1104-1120.

Smith, Robert P., and Tjandra, Primanata. "Experimental Observation of Iteration in Engineering Design." Research in Engineering Design 10 (1998): 107-117.

Smith, Robert P., and Morrow, Jeffrey A. "Product Development Process Modeling." Design Studies 20.3 (1999): 237-241.

Steward, Donald V. "The Design Structure System: A Method for Managing the Design of Complex Systems" IEEE Transactions on Engineering Management 28 (1981): 71-74.

Steward, Donald V. "Planning and Managing the Design of Systems." 91 Portland Intnational Conference on Management Engineering and Technology (1992): 189-193.

Tabors, Richard D., Hornby, Rick. "The Use of Multi-Attribute Trade-Off Analysis in Strategic Planning For an Electronic Distribution Utility: An Analysis of Abu Dhabi

Distribution Company." Proceedings of the 38th Hawaii International Conference on System Science (2005).

Teahen, John K. "Import Brands Steel First Quarter Thunder." Automotive News 9 Apr. 2007: 49.

Ulrich, Karl T. and Steven D. Eppinger. Product Design and Development 3rd ed. New York: The McGraw-Hill Companies, 2004.

Whitney, D. E., Mechanical Assemblies: Their Design, Manufacture and Role in Product Development, Oxford University Press, New York: 2004.

Yassine, Ali A. et al. "Do-it-Right-the-First-Time (DRFT) Approach to Design Structure Matrix (DSM) Restructuring." ASME 2000 International Design Engineering Technical Conferences. Baltimore, MD. 10 Sept. 2000.

Zambito, Antonino P., "Using the Design Structure Matrix to Streamline Automotive Hood System Development," SM Thesis. MIT Libraries, Cambridge, MA: 2000.