Platform Project Management:
Optimizing Product Development by Actively Managing Commonality

by

Raúl Pinillos Montaño

B.S. Mechanical Engineering, Universidad Nacional Autónoma de México, 2007

Submitted to the Engineering Systems Division
In partial fulfillment of the requirements for the degree of

Master of Science in Engineering and Management

at the

Massachusetts Institute of Technology

June 2011

Author: ______________________________

Department of System Design and Management
May 6th, 2011

Certified by: ______________________________

Olivier L. de Weck
Associate Professor of Aeronautics and Astronautics and Engineering Systems
Thesis Supervisor

Accepted by: ______________________________

Patrick A. Hale
SDM Director
Professor of Engineering Systems

Platform Project Management:
Optimizing Product Development by Actively Managing Commonality

by

Raúl Pinillos Montaño

Submitted to the Engineering Systems Division on May 6th, 2011 in partial fulfillment of the requirements for the degree of Master of Science in Engineering and Management

Abstract

Product platforms have proved to be an effective strategy for designing and manufacturing products in companies that provide different products for different customer needs. By designing common parts and creating product families, these companies have increased the profitability of their product lines leveraging economies of scale by increasing the volume of common parts and by sharing the development costs and investment among different products.

However, managing common designs in product family development is not a trivial task. Product commonality usually decreases over time, a phenomenon called divergence, usually present in the development of complex products like automobiles. Furthermore, all the products from the product family will be designed in a product development project; whether they are executed in a major project or in individual projects depend on the complexity and scope of the product. A usual practice has been to develop these products in different projects due to limited availability of resources, creating an additional challenge for managing these common designs because of their different lifecycles, usually contributing to increase divergence.

The main focus of this work is to understand the impact of designing a product platform on its associated development project(s) that share their resources including product components, facilities and human resources. The context of the study is scoped towards the dynamic nature of the execution of the project plan, rather than the product planning itself that is well covered by existing literature. To acknowledge the dynamic nature of the project, a system dynamics model that simultaneously simulated the lead and the derivative projects was developed based on their product commonality. The model was calibrated and complemented by a case study based on the development of a product platform in the automotive industry.

Divergence rates were measured and were found to range between 0.4% to 1.2% loss of product commonality every month. These typical divergence rates were included into the system dynamics model and were found to cause significant effects to the product development project which can be as high as a 22% schedule overrun or a 29% increase of the required personnel to achieve the planned project schedule. These significant effects to the development project caused by non beneficial divergence should be avoided, concluding that actively managing product commonality can be an effective method to achieve a successful execution of the development projects when the product platform approach is utilized.

Thesis Supervisor: Olivier L. de Weck
Title: Associate Professor of Aeronautics and Astronautics and Engineering Systems

Acknowledgements

This is the end of quite a long winter in Michigan; besides the end of the freezing season, a new season is starting in my life. These are the final words I am writing to wrap up a work that has taken almost two and a half years since I started this amazing adventure at MIT. I consider myself fortunate for not only the life changing opportunity of studying at MIT, but also for the great experience of living abroad for more than three years. Above all, I am thankful for being surrounded by people who love me and have inspired me to take this quest.

It will not be possible to list all the people who have touched my life who in some way are part of this work. I would like to apologize in advance to anyone who is omitted here, YOU have contributed to make this happen and YOUR contribution is invaluable. Thank you for being part of my life. In particular, I would like to express my appreciation for the specific contribution of the following people:

> To my mom, Josefina, whose memory I would like to honor. Her love and determination have molded me into the person I am today.
>
> To my dad, Raúl, whose support throughout these two years has been unconditional and has encouraged me to give the best of me everyday.
>
> To my siblings, María José and Jaime, for inspiring me and cheering me up to continue this quest. I really admire both of you.
>
> To my grandmother, Fina. She will never fill the emptiness for loosing my mom, but she has helped me to make it barely noticeable.
>
> To my work and prior SDM colleagues: Adrián, Takahiro (we miss you), Adrián, Antonio, Marcia and Antonio and for making a more challenging experience for future generations and sharing several insightful thoughts with me.
>
> To David, my relentless company throughout this journey both at work and in SDM. It has been an honor working with him and sharing the joy and pain of the SDM program. I will never forget those endless nights towards the end of each semester when deadlines were approaching
>
> To my SDM09 and SDM10 colleagues. The experience at MIT would not be complete without excellent students like you, especially to Mario, Jaime, Matt, Charlie and Jui for sharing their friendship and time with me.
>
> To my three sponsors: SDM, CONACYT and my Company, for making a dream come true. I will always be on debt with these three great organizations, a debt that goes beyond money.
>
> To the SDM staff and the MIT faculty, for making this an unforgettable journey and shaping a new systems thinker; especially to Pat hale who has believed and supported the close relationship between SDM and my Company.
>
> To Olivier de Weck, for allowing me to see Project management with completely new eyes and for supporting this work. Every communication I had with him regarding this thesis was full of positive thoughts and challenging questions which encouraged me to continue this quest.
>
> To my work colleagues who have also had the opportunity to live abroad in Michigan: Armando,

David, Mauricio, Alejandro, Ivan and Carlos. Living away from my family can be a hard experience, but friends like you not only make it easier, but enjoyable. Special thanks to Carlos who has been a great support in Six Sigma related inquiries.

To Peg, for investing so much time in reading this work and helping me with my English skills

To Michael, Tim, Todd, Brett, Willy, Anand and Jeff, for pioneering the concept of Platform project management in the company and inspiring on my thesis topic. .

To the rest of project team members which provided some key information for the case study that was one of the key foundations of my work.

To my Company's senior leadership team; especially to Marcos, Alejandro and Raúl for supporting my development inside the company, improving my business and leadership skills and for trusting in capacity to get my Masters degree.

To Enrique, for sharing his wisdom and for acting as my internal thesis advisor within the Company. Thank you for supporting the development of my thesis topic, for your guidance during these two years, and for exposing me to greater challenges I would never imagine.

To my family in law who has encouraged and supported me in such a meaningful way by giving up Hanzel to be with me. I know how it feels when she is far away.

And last but not least to my wife, Hanzel. I will never have enough words or acts to show my appreciation for standing up with me every second I have invested in this work. I really admire your inner strength; you are the reason I wake up every day... You are simply wonderful. The only think I can imagine is better than getting a degree from MIT is being married with you. I do not know what the future is planning for us, the only thing I know is that I will be with you and that makes it a great future. Thank you.

Table of Contents

List of Figures

Chapter 1

Chapter 3

Chapter 4

Chapter 5

List of Tables

Chapter 2

Chapter 3

Chapter 4

Chapter 5

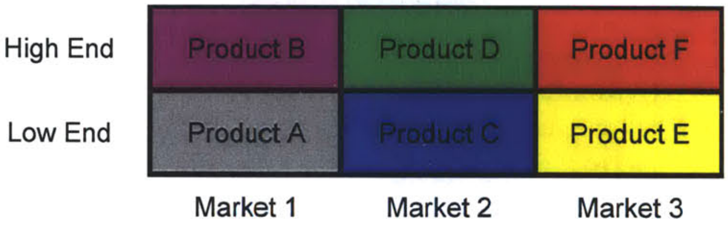
1. Introduction

1.1 Problem Statement

A continuous challenge for product development is to develop more products better, cheaper and faster. Product platforms have emerged as an alternative that could meet all these goals. By commonizing designs, companies using the product platform approach have achieved significant improvements in product cost, design lead time, and better quality as found in examples in the electronics (Sanderson, Uzumeri, 1995), power tools (Meyer, Lenherd, 1997), and automotive industry (Cusumano, Nobeoka, 1998). Nevertheless, most authors agree that the main risk of commonizing these designs is not achieving enough differentiation between the products, and customers could potentially criticize the firm for this lack of differentiation. An extreme example of this situation has coined the term "badge engineering" in the automotive industry, referring to the negative connotation of excessive design reuse across vehicles that should not be confused with the platform approach.

A product platform is defined as a "set of subsystems and interfaces that form a common structure from which a stream of derivative products can be efficiently developed and produced" (Meyer, Lenherd, 1997). From now on it will be referred as "platform" to the "product core" (Meyer, Utterback, 1992) or in other words, the core technology or common building blocks that can build two or more products. However, this concept has been broadened beyond the product core to include also processes, interfaces, infrastructure, human resources and knowledge that is shared by a set of products (Robertson, Ulrich, 1998).

A "product family" will be understood as the set of products that are built from the same platform or product core. Each of these products is called a "platform derivative". The figures below are graphical representations of the product family that are commonly used in product platform literature, the first one (figure 1.1) is a generic market segmentation matrix and the second one is a product family map (figure 1.2), which plots the relationship between products in a time scale.



High End	Product B	Product D	Product F
Low End	Product A	Product C	Product E
	Market 1	Market 2	Market 3

Figure 1.1 Market segmentation matrix. Each product is a platform derivative (Adapted from Meyer, Lenherd, 1997)

The design and development of each of the platform derivatives will be executed in a product development project. It is up to the firm to decide how to group the development of these derivatives into one or a set of projects. This idea may be easily represented in the product family maps. Figures 1.3a to 1.3c detail three examples of how the derivatives from the market segmentation matrix could be organized in different development projects. Regardless of the project to product mapping strategy, one can acknowledge the existence of a higher level project that comprises the development of the entire platform of derivatives, and this project will be called the "Product Family Project" and will be defined as the set of all the projects or tasks to develop all the derivatives to be designed from a common product platform.

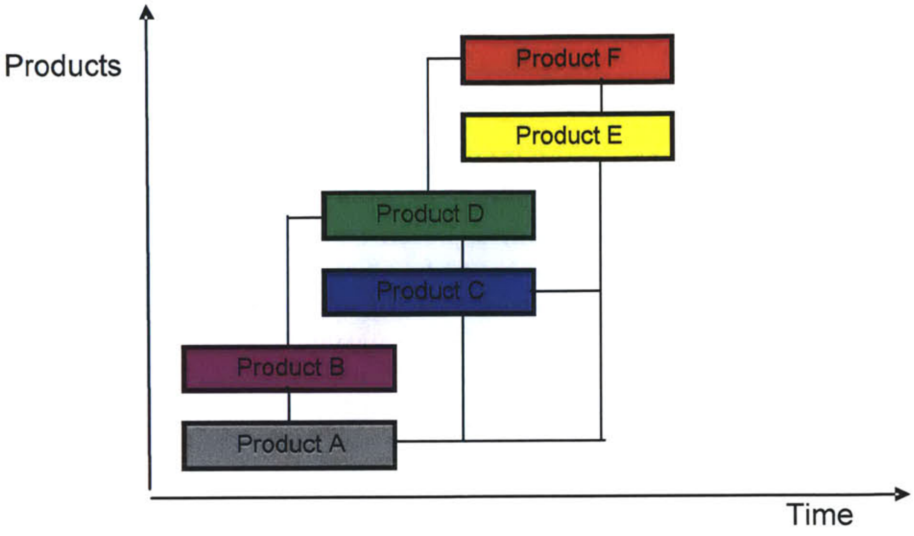


Figure 1.2 Generic Product Family Map ("Product Plan" adapted from Robertson & Ulrich, 1998).

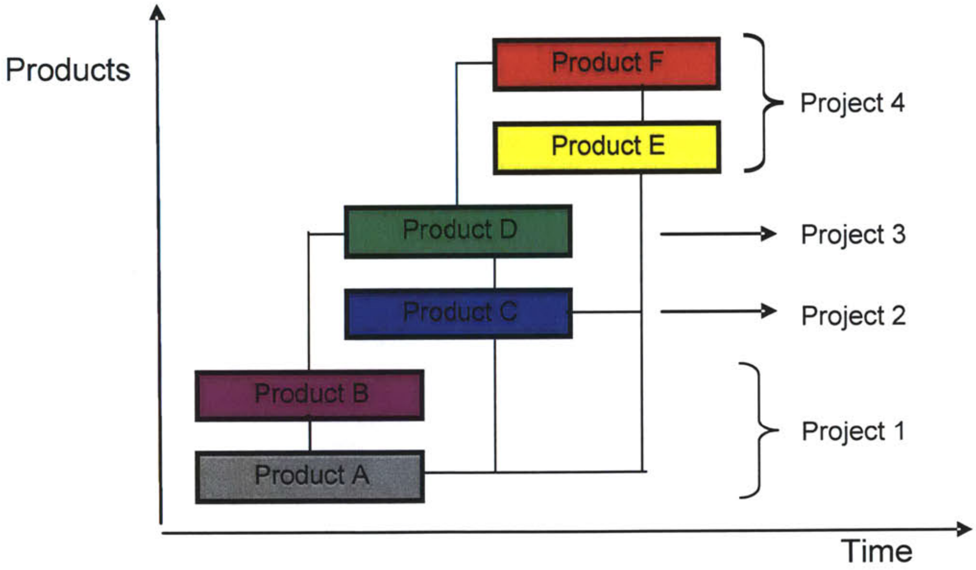


Figure 1.3a Product to project mapping – Example 1. Some projects share same timing.

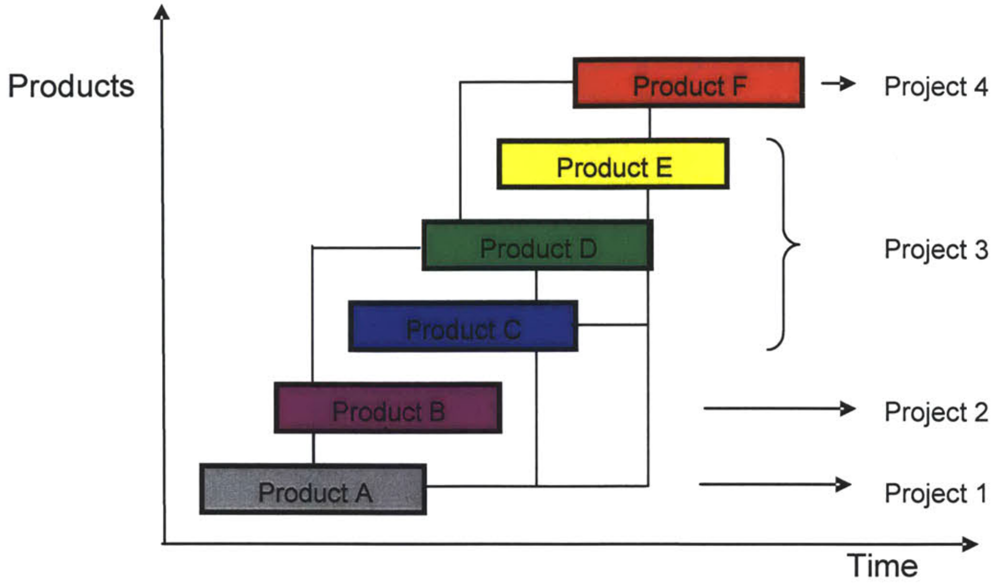


Figure 1.3b Product to project mapping – Example 2. All projects differ on timing, but they can still be grouped according to the organization's preference.

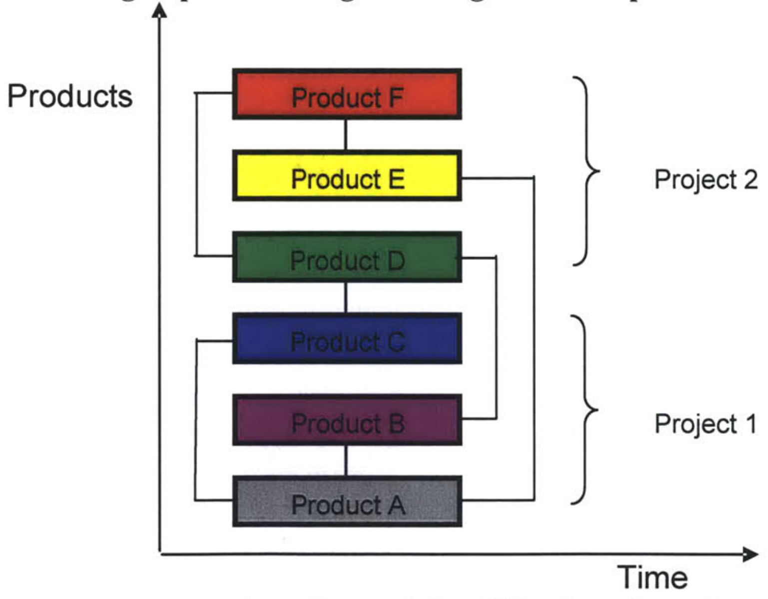


Figure 1.3c Product to project mapping – Example 3. All Products have the same timing, but they can be grouped according to the organization's preference.

The way projects are mapped to the products can have a deep impact to the product family project. There may be reasons to group the products in projects that have similar timings, and are built in the same manufacturing location or classified by market segment or how similar they are among each other; it all depends on how the major product development stakeholders (design, manufacturing and marketing) understand the similarities of each of the projects. Figure 1.4 is an example of this concern. If the projects are categorized by marketing, they could see either two market segments, three marketing regions or a total of five different products (market segment and region). If the design organization categorizes the projects on how similar the designs are to each other, they could see either four major styling variants or six design variants in total. Finally, if manufacturing categorizes the projects, they could see either three manufacturing locations affected or five different production start dates.

Marketing			Design		Manufacturing	
Market Segment	Regional Market	Market segment + region	Styling	Sub Variant	Manufacturing Location	Launch Date
A	A	A	A	A	A	A
A	B	B	A	A	B	B
A	B	B	A	B	B	B
A	C	C	A	A	B	B
A	A	A	B	D	A	C
B	A	D	C	E	C	D
B	A	D	C	E	B	D
B	B	E	C	E	B	D
B	A	D	D	F	C	E

Figure 1.4: Example of a project segmentation according to the major product development stakeholder's criteria.

To unify the criteria, an acknowledged project within the company that will have its own budget and business case will be called a product family subproject or "derivative project". The criterion is selected for two reasons: firstly, because there should be some corporate oversight on what makes sense to track separately and what does not. Secondly, usually all the shared resources charges are prorated among all the affected entities and allotted to the first project that needs to make the expense; and, all the unique expenses will be allotted to each of the corresponding affected project. As seen in the matrix above, as more derivatives or markets are added to the product family, it makes its financial management more complex (what part is shared within which products); however, the required effort and project scope are usually studied as an incremental opportunity to whatever has already been planned. To simplify the framework, it will be assumed that every selected project entity will develop a corresponding derivative product (or group of products), so we may use project and product in the same context, as represented in figure 1.5.

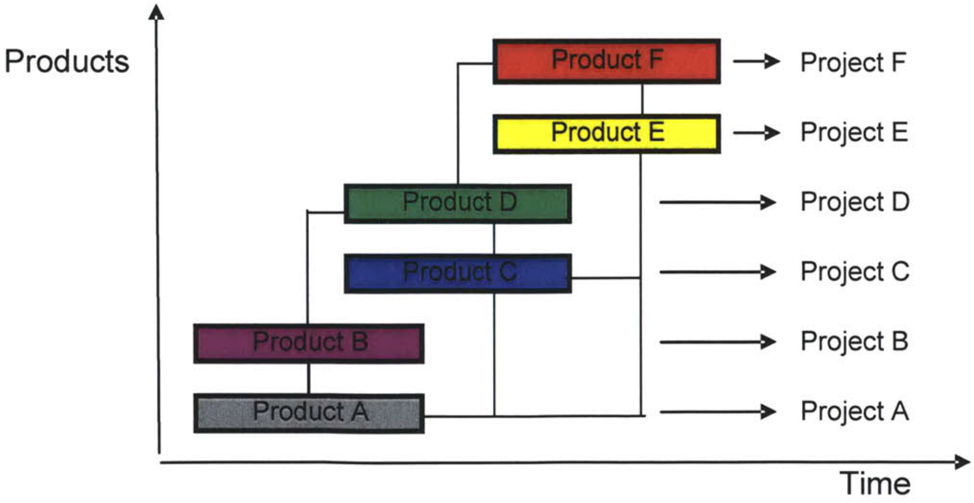
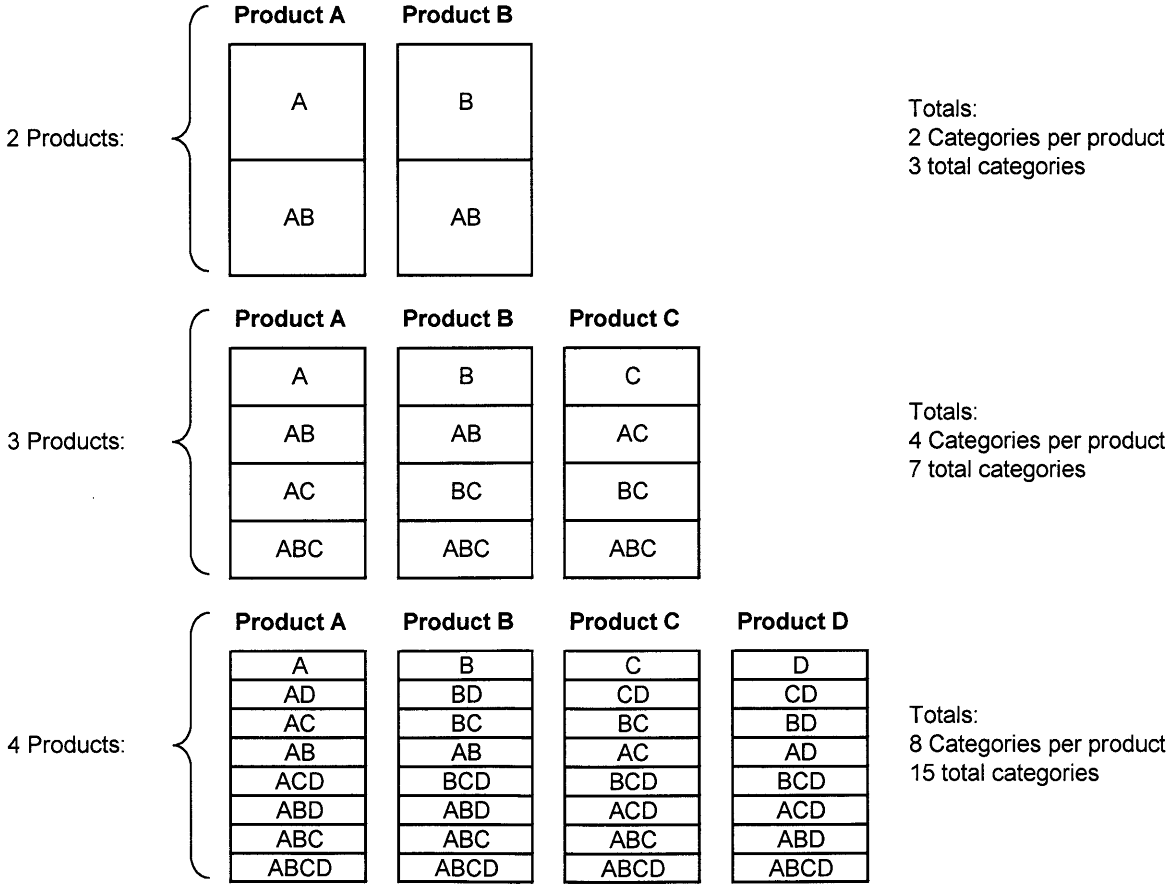


Figure 1.5: Simplified project to product mapping to be used in this dissertation. Every product (with its minor variants) will be developed by a different project.

The financial management of the projects and the "physics" of what should be designed first relative to the other projects creates the concept of the "lead product", "lead project" or "lead derivative", that will be defined as the first product/project to be introduced to the market relative to the others. In other contexts, the first derivatives are also known as the "platform" (Muffatto, 1999), but this work will use the term lead derivative instead to avoid confusion with the definition of platform as the product core. The concept of the "lead" project or derivative is important when a team is developing the product platform since many of the common components will be designed for the first platform derivative and reused for the "follow-on" derivatives. Boas (2008) provided a refined research on the industrial practice of the development of product families and acknowledged that complex product families are developed in a sequential instead of a parallel manner.

The example above is a clear illustration of the interrelationship between product commonality and the corresponding projects for each product (which are now mapped one to one). As products are developed, the "lead" product becomes the baseline and the follow-on products are incremental applications added to the product family; this is consistent with the mental model of the product family stakeholders where the additional products to the product family are studied by identifying the incremental unique development. However, as more products are added to the product family, the parts sharing assumptions become exponentially more complex as detailed in figure 1.6. In the figure, the letter or set of letters in each box represent the products in which that part is used. For example, "A" means that it is used only in product A, "AB" means that the part is used in both product A and product B.



Product A	Product B
A	B
AB	AB

Product A	Product B	Product C
A	B	C
AB	AB	AC
AC	BC	BC
ABC	ABC	ABC

Product A	Product B	Product C	Product D
A	B	C	D
AD	BD	CD	CD
AC	BC	BC	BD
AB	AB	AC	AD
ACD	BCD	BCD	BCD
ABD	ABD	ACD	ACD
ABC	ABC	ABC	ABD
ABCD	ABCD	ABCD	ABCD

Figure 1.6: Sharing complexity. Each product is built by the combination of unique and shared components. Each of the shared components is detailed with which products it is shared with. As more derivatives are added, more sharing categories are created.

Derived from figure 1.6, using combinatory, a general relationship between the total number of categories and the total categories per product can be derived.. If "n" is the number of derivatives, to calculate the total number possible combinations, we use the general formula for combinations where n is the number of things to choose from (the number of derivatives) and r is the number of derivatives chosen where r is equal or less to n:

$$Number\ of\ possible\ combinations = \binom{n}{r} = \frac{n!}{r! \cdot (n-r)!} \tag{1.1}$$

To add all the different sizes of things to choose from, the total number of derivatives can be estimated using the following relationships:

$$Total\ Categories\ (Tc) = \sum_{r=1}^{n} \binom{n}{r} \tag{1.2}$$

To only account only for the combinations of one product, the formula above is updated as follows:

$$Categories\ per\ Product\ (Cpp) = \sum_{r=1}^{n} \binom{n-1}{r-1} \tag{1.3}$$

However, these relationships can be further simplified by using the following formula (Source: Wikipedia: Binomial coefficient)

$$Sum\ of\ all\ combinations = \sum_{r=0}^{n} \binom{n}{r} = 2^n \tag{1.4}$$

Therefore, the general relationship of the number of total categories (Tc) and categories per product (Cpp), based on the number of derivatives will be:

$$Categories\ per\ Product\ (Cpp) = \sum_{r=1}^{n} \binom{n-1}{r-1} = \sum_{r=0}^{n} \binom{n-1}{r} = 2^{n-1} \tag{1.5}$$

$$Total\ Categories\ (Tc) = \sum_{r=1}^{n} \binom{n}{r} = \sum_{r=0}^{n} \binom{n}{r} - 1 = 2^n - 1 \tag{1.6}$$

As seen in equations 1.5 and 1.6, the complexity of integrating products in the product family is exponentially complex. Even if a significant proportion of the product can be shared for two or more products, it does not create significant efficiencies to product integration. For every new product added to the product family, the product family complexity grows geometrically per the equations 1.5 and 1.6 and represents another challenge in product platform development.

The products' components, parts, modules and systems sharing assumptions are not static. As the product development project progresses, some inconsistencies or issues can be found in the original commonality/sharing plan, imparting a dynamic nature to the project. As these issues are brought up, they have to be solved and a possible solution is to decrease the expected commonality between the products over time to create unique solutions for the derivatives. This phenomenon is called "divergence" – the loss of commonality over time (Boas, 2008). Divergence is a common phenomenon in complex product families (i.e. Aerospace, automotive, satellites, etc), understanding "complexity' as the total number of parts in these products. The opposite phenomenon is called "convergence", where product commonality increases as opportunities are found during the development project. Usually divergence issues appear more often than convergence opportunities, so the final result for the product is a reduced commonality scope than originally planned (Boas, 2008). Most of the literature in product platform is biased towards platform planning and commonality plans, however, the dynamic effect of the development of these platforms is not as widely addressed in the literature.

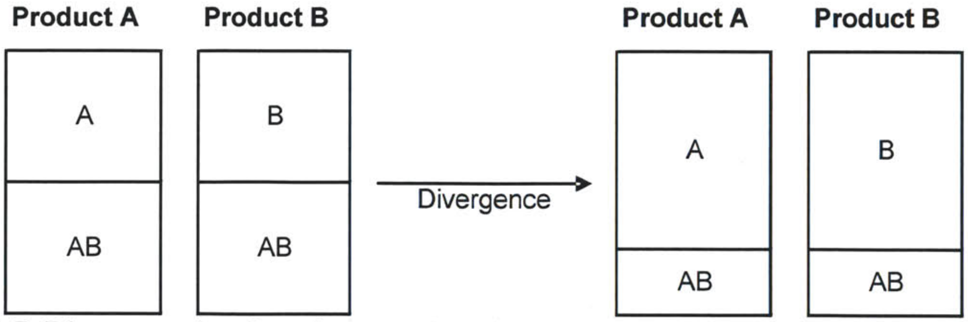


Figure 1.7: Divergence in a 2 variants product family example. The original product sharing assumptions was that 50% of the parts will be shared within A and B (therefore the 50% remaining would be unique parts for A and B) after the products were developed, the number of shared components decreased (to 25%) and the number of unique components increased (to 75% each).

Products are intimately related to their projects, this means that divergence will have also an impact not only to the products but also to their development projects. When a previously assumed shared part can not be shared, regardless of why it cannot be shared anymore, a new part will have to be designed, tested and tooled, and this might impact the project's budget, resources and schedule. In this scenario, the projects are creating rework for each other as they will not be able to be developed as originally planned. This issue is even worse considering the mental model of how product derivatives are planned, since a follow on project is relying on the lead project to design the common components, creating a new workstream that was not originally planned. This issue is also recognized in the literature as lead products usually need more development time than the follow on derivatives (Meyer, Lenherd, 1997).

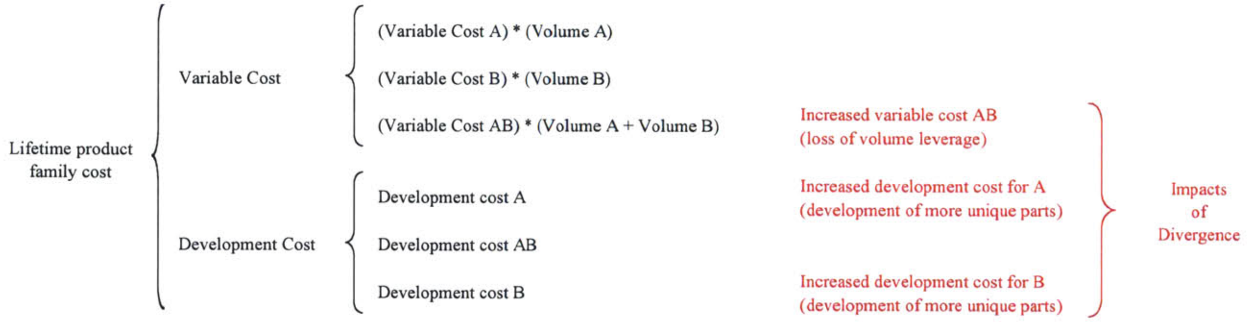


Figure 1.8: Impacts of divergence. When the sharing plan is not achieved, lower economies of scale and higher development costs will be expected.

It should be noted, however, that in some cases divergence should be allowed – even given its cost penalty – in order to preserve the uniqueness and competitiveness of each variant. It is when divergence occurs for purely organizational reasons that value may be lost. As stated previously, the product and the project are dynamic entities. While most of the literature on product platforms is focused on the planning portion of the product and the project (Meyer, 1997; Robertson and Ulrich, 1998; Bowman in Simpson et. al, 2006), they do not mention the importance of the execution portion of the project, also called project control, which involves the "coordination and facilitation of all the tasks required by the project and the project adaptation to the issues found during the development of the product" (Ulrich, Eppinger, 2008). During the planning portion, the product family maps and the product architecture are settled; the platform is understood as a whole and the process is mostly static. However, once the product platform plan is laid out, then the execution of the product family plan is usually done in phases in order to overcome existing resource constraints. Ideally, the best way to avoid divergence is to develop the product platform for all the derivatives concurrently; however, this usually is unfeasible in terms of the required resources. The phenomenon of developing the derivatives in a staggered manner is called "lifecycle offset" (Boas, 2008). Figure 1.9 is a graphical representation of the lifecycle offset concept.

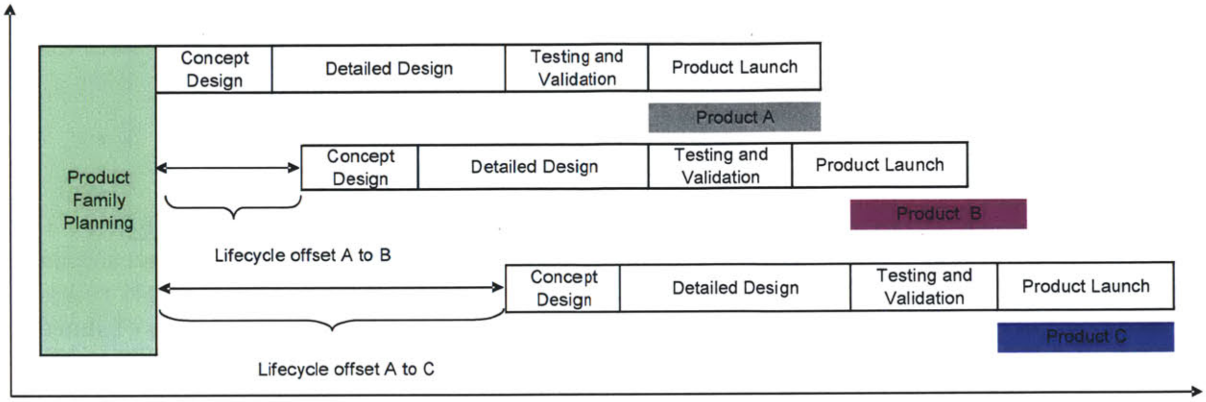


Figure 1.9 Lifecycle offsets and product development process in a product family map (adapted from Boas, 2008). Sometimes only the product planning phase is done concurrently for all the derivatives.

1.2 Platform Project Management

One of the drivers for divergence is the lack of coordination among projects (Boas, 2008), as they may be driven by different teams and project managers. As the product family becomes increasingly complex, a new entity is required to ensure that the development is performed as planned, to reduce the divergence of the product platform and to manage the lifecycle offsets of the derivative projects. If every change and decision to a common component is studied considering all the derivatives, regardless of what derivative drives the change (lead or follow on), then the probability of having divergence in the product platform should be reduced. A change to a common part should be avoided if it is not beneficial to the product family overall. This organizational entity will balance the needs of the common components across projects and will drive its decisions based on the overall product family project impact rather than a derivative project in isolation. In some cases some derivatives may carry more weight than others due to their anticipated sales volume or contribution to overall profitability. This entity will be called the "platform project" and will be lead by the "Platform Project Manager", although other terms could be used to identify the key stakeholder to manage the product family like "Platform Director" or "Core Platform Manager" (Mahmoud-Jouini, Lenfle, 2010). The term "Platform" will still have the same connotation of the product core. Figure 1.10 lists some of the elements of the Platform Project.

Platform Project	Objective	Maintain product commonality by balancing decisions across multiple products and projects and by coordinating shared resources across different products/projects
	Leader	Platform Project Manager
	Scope	Product Platform – The shared core technologies among different products
	Tasks	Product development process tasks for the elements of the core technologies of the product
	Resources	All resources related to the execution of project tasks, with increased relevance to shared resources for different products
	Schedule	Starts with the development of the first product and ends with the market launch for the last derivative

Figure 1.10: Elements of the platform project

Platform Project Management will be defined as the "set of activities related to the design and development of the product platform and the coordination of the development projects within a product family based on a common platform." The activities of the platform project manager do not differ too much from usual project management, as they also have to balance the "iron triangle" of the project: scope, cost and schedule; however, when managing a portion of several other projects, some differences are found. The iron triangle framework will be used to highlight the main differences of platform project management and conventional project management. The platform project iron triangle will be based on how the projects share parts, processes and resources:

- Project Scope: Scope is reduced to core technologies and common/similar components instead of the overall product. Project scope could be managed through incremental product commonality and uniqueness.
- Project cost: Development and investment costs should be only costs related to the core technology and its interfaces. Also, the platform project manager should manage resources (facilities, human resources) that are shared among projects.
- Project Schedule: Project schedule will be understood as the time to develop all the core subsystems for all the different platform derivatives, but also has to manage the individual schedules for each of the projects, including the lifecycle offsets.

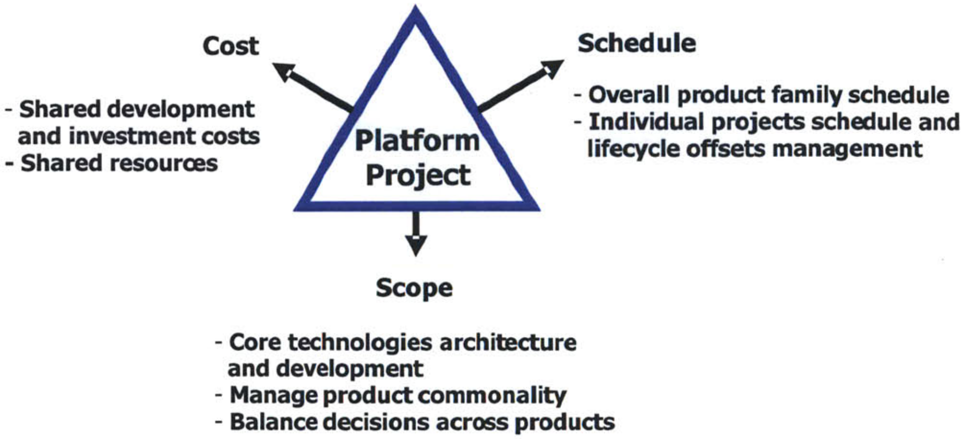


Figure 1.11: The platform project iron triangle

The product family is "technologically interconnected" (Cusumano, Nobeoka, 1998) by the product platform: a common architecture and common/similar parts. Comparable to how product development is usually organized in departments based on the product decomposition (Allen, 1984), the product family project should also have a governance structure that should mirror the interconnections existing in the product. The proposal is to make a division in the product architecture and identify the subsystems that will be considered part of the core technologies that should be reused (that will be known as the platform subsystems, parts and components) and the subsystems that should be differentiated (that will be known as the "differentiation subsystems"), similar to the proposed "commonality plan" and "differentiation plan" as proposed by Robertson and Ulrich (1998). This proposal to identify the common and differentiated parts is consistent with the way the Bill of Materials (BoM) and the organization are mapped to the product architecture. Figures 1.12a to 1.12f below show a graphical representation of this proposal:

1) Let product A, B and C be derivative products developed from the product family. Each of the products will be executed by a different product development project.

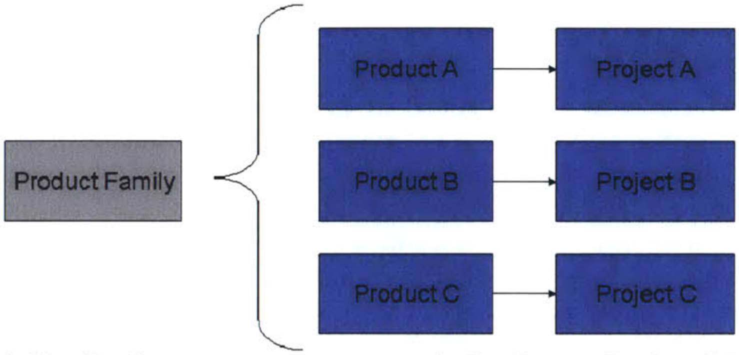


Figure 1.12a: Project governance proposal. Product to Project Mapping.

2) The parts existing in the bill of materials for each of the products can be broken down in two major categories: Platform Components (PC_X) and differentiation components (DC_X) parts for each product X.

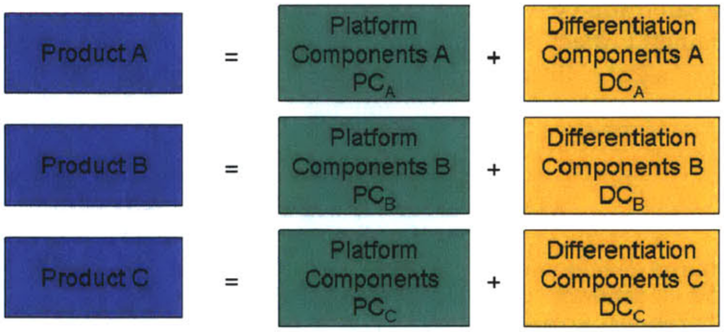


Figure 1.12b: Project governance proposal. Product breakdown.

3) The diagram above can be further refined by categorizing the parts according to their sharing strategy as "unique" or "shared" consistent with figure 1.6. To simplify the diagram, all the shared parts are included in the same subset, regardless the number of products sharing that part, although, as stated previously, adding more derivatives to the product family grows the number of categories exponentially.

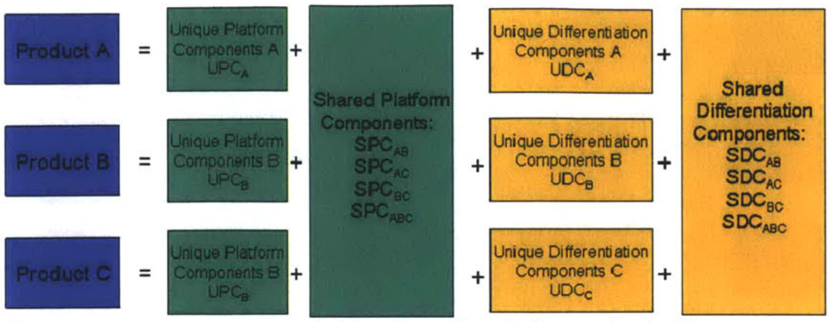


Figure 1.12c: Project governance proposal. Product breakdown including components sharing.

Figure 1.12c is related to the framework developed in figure 1.6. All the components with a sub-index with a single character (A, B or C) are the unique components, and all the components whose sub-index has two or three characters are shared components. Nevertheless, figure 1.12c further divides these shared components into the platform components, which should be mostly shared, and the differentiation components, which should be mostly unique.

4) As done in step 1, where each product is mapped to a project; a proposal could be to breakdown the project into two categories, the platform project (PP_x) and the differentiation project (DP_x) for each product X.

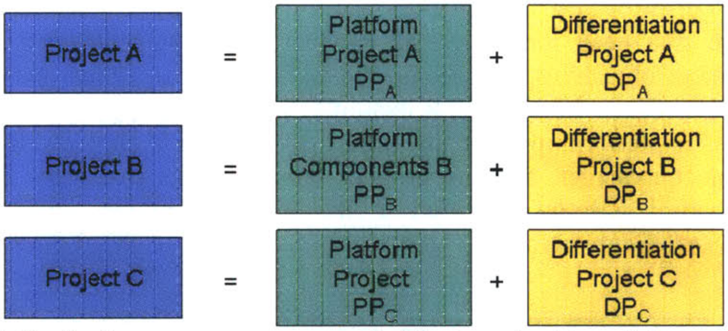


Figure 1.12d: Project governance proposal. First project breakdown proposal.

5) However, the above structure lacks an overall coordination and integration activity for each project and therefore would not be appropriate. Since the platform components should be mostly shared and the differentiation components should be mostly unique, the projects will be organized in an overall platform project that will oversee all the platform components and a differentiation project that will oversee the development of the differentiated components as well as the integration with the platform components. The figure below integrates the product structure with the proposed project structure; there is an overlay between the differentiation project and the platform project.

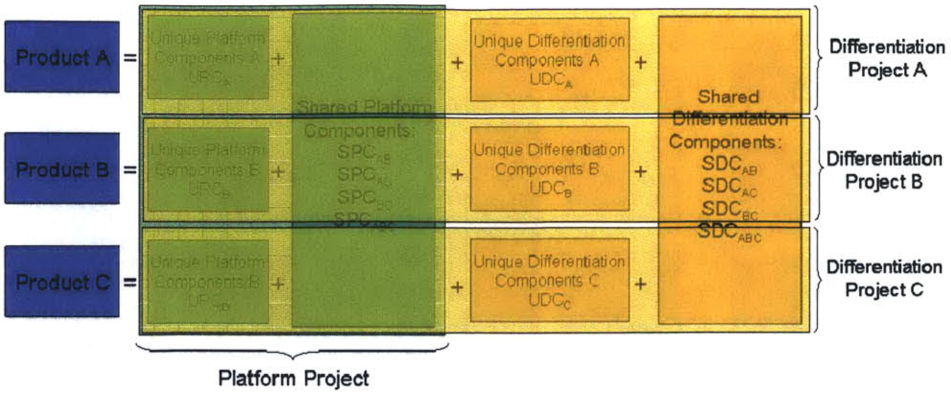
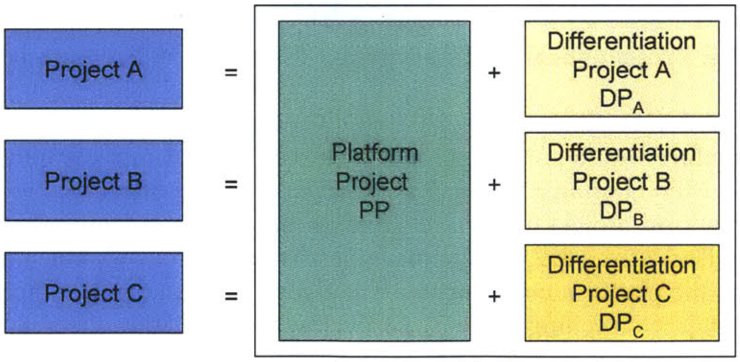


Figure 1.12e: Project governance proposal. Platform and differentiation projects proposal.

As explained before, the differentiation and the platform components can be either unique or shared; however, one would expect that the platform components should be shared and the differentiation components should be unique. When the opposite situation occurs, it creates a challenge to the organization proposed above. This is especially tough when the differentiation components are shared, which represents a potential risk to the product's acceptance in the market because sharing the parts within the different products will be evident to the customer. The organization above should try to avoid designing unique designs for the platform components and shared designs for the differentiation components to make the proposed organization work.

6) Finally, there is a need for an overall product family stakeholder to resolve the conflict of opposing project objectives and decide the tradeoffs (cost, scope and schedule) as an overall product family project. In this simple example, the resulting project structure and governance are four projects: a platform project and three differentiation projects which will be lead by a different project manager, and the product family project manager. The preliminary project structure proposal will be modified to the proposed project structure and governance as represented in the figure below.



In gray: Overall Product Family Project PFP

Figure 1.12f: Project governance proposal. Platform and differentiation projects proposal with overall product family project.

The proposed project governance and structure gives some advantages when the products are being planned and designed as a product family:

- The tension between what parts and components to commonize or to differentiate might be solved by the structure; since there is a different stakeholder that will be motivated by a different priority: commonization vs differentiation. It is recommended that the projects (A, B, C) be managed separately to have enough differentiation within them. When the opposite situation happens and the same project manager is assigned to several projects, the challenge is having the same level of commitment to all the projects as found by Cusumano and Nobeoka (1998).

- The structure is based on the product architecture; this should enable a better communication within the project. The structure of the projects follows product architecture and the organizational structure also follows product architecture.
- The scope for each of the project manager's activities is reduced and enables greater focus.
- The platform project manager will naturally handle the coordination among common components across projects, as the effects for all the derivatives will be under his/her scope and will hold him/her accountable to all the projects. Decisions will be made in the perspective of the overall family and not a specific variant, and should help with the "single product mindset" issue.
- If organized as a separate project, the platform components could be designed prior and in relative isolation to the differentiation components.

1.3 Research questions

The concept of the "platform project" has been formally introduced, however, given that the terms platform and product family are usually used in the same context, to avoid confusion between both terms, these concepts can be understood as in figure 1.13. If building a pyramid from bottom to top, each of the supporting blocks are subsets of the block above. This means that a product portfolio in a company is built from different product families, and each product family is built from product lines and each product line from its product variants and so on. Walking from a product perspective on the upper pyramids to a project perspective on the lower pyramids, the right upper pyramid (which is the same as the left lower pyramid) introduces the concept of the platform as a subset of each product, which is built from core shared technologies or subsystems, modules, parts and components (potentially including software). The effect to their corresponding projects is represented as the rightmost pyramid, where a platform project and a differentiation project exist, but they could be broken down by each of the subsystems projects.

In conclusion, the scope of the present work is to develop a framework to understand the nature of platform product development projects in the context of a previously planned product family that has to be executed and adapted to the inherent uncertainties and risks of product development. The key research questions in the present thesis are:

1) *What are the implications to the development project(s) of having products that are developed from a common platform? What are the synergies and unwanted effects of multi project coordination based on product commonality? What are the effects to the overall product family project and the individual projects?*

2) *What are the reasons and the effects of product divergence, or loosing product commonality over time? What are the most relevant countermeasures to achieve all the project's objectives?*

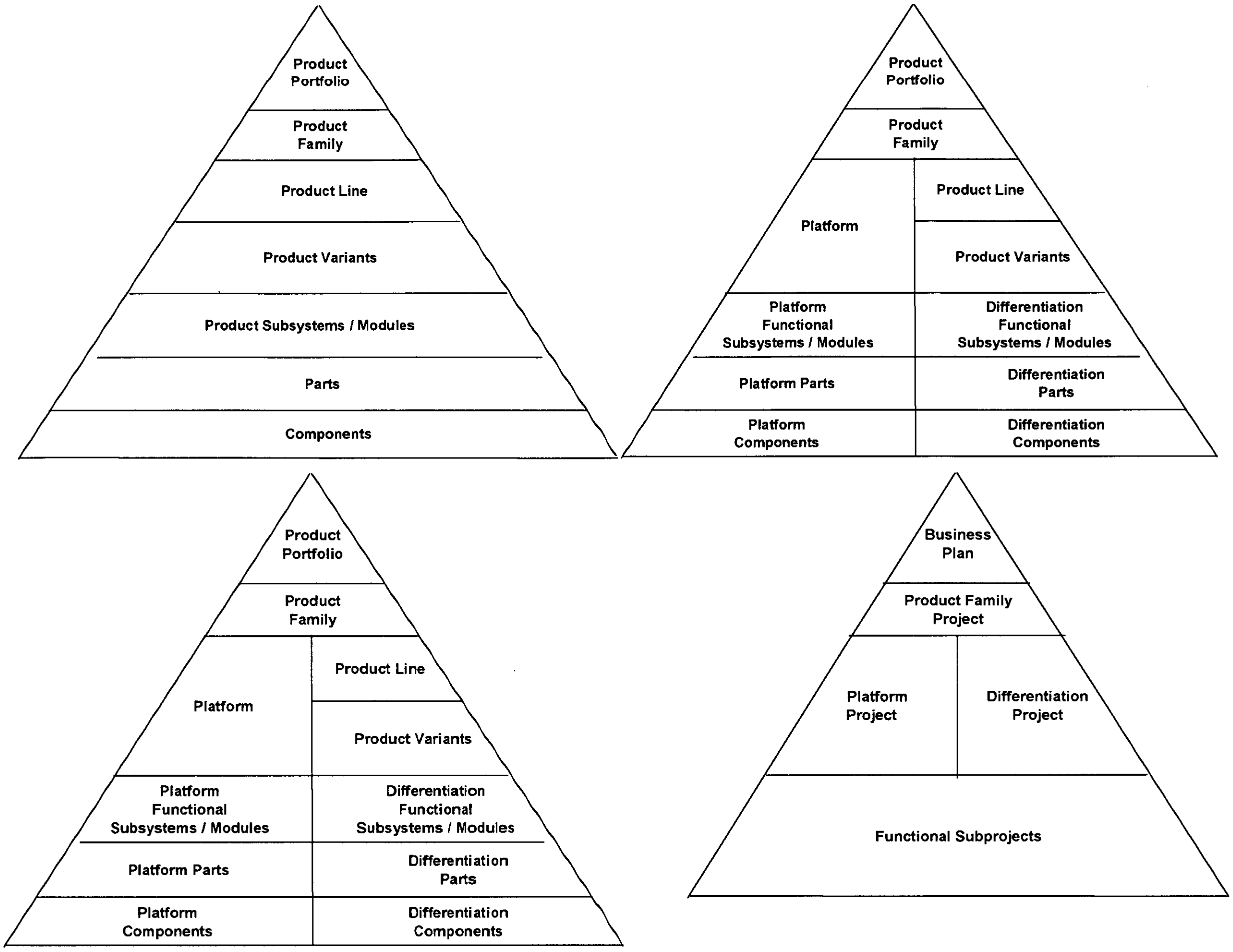


Figure 1.13 Relationship between the products, product architecture (platform and differentiation) and their projects.

To solve the questions above, this framework will be analyzed using system dynamics to simulate the effect of the design changes on the affected product development projects in terms of the project scope, costs and schedule. This study will be complemented and tested with a case study based on an actual automotive platform project. The intention of solving the questions above is to provide a set of managerial recommendations and insights on how to organize and execute product family projects and how to overcome their main challenges: the balance between product commonization and differentiation, the dynamic nature of the product development process and the risk of product divergence, the coordination among projects that may have different timelines and shared resources, avoidance of the single product mindset while making tradeoffs within multiple project stakeholders, and the complexity of managing concurrent projects.

2. Review of Relevant Literature and Proposed Framework

2.1 Review of Relevant Literature

The knowledge of product platforms and project management can be found in either of their numerous related literature sources; however, it is uncommon to find both pieces of knowledge together. The literature on product platforms is centered on the product itself and the product development process, and some authors also acknowledge the impact that the platform development creates on the development project and the team organization. On the other side, the project management literature is usually centered on the management of single projects and some work has been done on understanding the management of multiple projects in a coordinated way known as "Program Management" (PMBOK, 2008).

This literature review chapter aims to describe the most relevant sources within the context of product platform development. Specifically two key areas were researched within the product platform approach: the literature in project management, where the key variation in a product family context is the execution of multiple projects for the each of the products of the product family; the second key area researched was product commonality and product variety, since they are key enablers or measures to design a product family.

Finally, two key tools are used to understand the impact of product commonality and multi-project management into the context of platform product development: system dynamics and an exploratory case study of the development of a product platform in the automotive industry. A brief stream of literature research was performed on these two key aspects of the present dissertation to understand the benefits and limitations of the system dynamics method, as well as the comparison of the findings in the single case study compared to other examples in the same industrial environment.

2.1.1 Product Platforms and Product Families

The literature in product platforms and product families has become one of the most important areas of research in product development in the last 20 years. The term "platform" has evolved significantly from product platforms understood as common components and design reuse, to a broader perspective to include all the assets shared by a set of products. In an even broader scale the term "industry platforms" has been developed (Cusumano, 2010) to refer to all the complementary products and services and network effects around a product, however, the scope of this work is around the product platforms. The developed knowledge is vast ranging from the description of the original idea based on successful industrial examples of product platforms and product families to a very detailed set of methods for their design and implementation and more detailed frameworks on their benefits and their unfavorable effects.

Meyer and Utterback (1992) pointed out that while the prior practices were focused on developing single products as fast as possible; it had redundancy in the technical and marketing effort and a lack of long term consistency. They argued that developing product families should be the focal point of attention and product family planning should be considered a core capability of the firm. They provide a clear distinction on the concept of the product platform as the shared design and components from a set of related products and the product family as the set of these products sharing the same platform or "product core". They also introduced the concept of the product family maps, which represent not only the products associated with the product platform but also their evolution over time. Finally they conclude that the firm should focus on enhancing the product core (the platform) as a key for sustained success of the overall firm. While their work provides a useful framework for platform planning and platform renewal they do not detail how to execute these plans in their corresponding development projects, besides acknowledging that senior leadership is required to cascade the product family vision and adapt their budgeting for these means.

Meyer and Lenherd in "The Power of Product Platforms" (1997) also agree that a company's long term success resides on a continuous stream and set of products instead of a single product. Furthermore, they affirm that by planning and developing a product family as a whole, it can be efficiently created from the foundation of a common core technology. Their work provides a methodology and strategy for designing, developing and renewing a product platform over time. The proposed framework for developing product platforms / product families: "the power tower", includes understanding the market applications, defining the platform – the common building blocks -, defining the architecture and interfaces, developing the rollout plan of the other derivatives and building the implementation team. The audience of this work is targeted to product development management and provides a series of benefits of designing a product platform based on the successful experience of the Black and Decker power tools and the Xerox company. While Meyer and Lenherd present a series of applicable principles for product family planning, their work does not provide recommendations for the implementation of these plans and the dynamic nature of product development. While product and platform planning is essential for success, a flawless execution of the plan may be as important as the plan itself.

Ulrich and Robertson (1998) broadens the concept of the product platform not only considering the core technology, but as the collection of assets shared by the set of products, and may include components, knowledge, production processes and people and resources. They acknowledge that the key issue with platform product development is providing high commonality to achieve lower costs with proper product differentiation to ensure market success. To achieve the diverse customer needs while using common assets for their development they present an iterative planning framework that consists of three closely related plans that must be consistent with each other: the product plan, the differentiation plan and the commonality plan. By illustrating their method with the design of an automotive instrument panel, they acknowledge that the product architecture will greatly influence the nature of the tradeoffs between commonality and distinctiveness that will be both understood as an important attribute of the product. While they create a framework for introducing commonality as a key driver for platform development efficiency, their scope is limited to the early phases of product development and reduced to the selection of the product architecture. Nevertheless they provide a useful framework for decision making based on the products profitability along with market success that could be used for future design changes and decisions.

Finally, a very complete anthology on the current progress on product platforms and product families has been summarized in an edited volume by Simpson, Siddique and Jiao (2006). They categorized the current knowledge in three main areas being product definition, product design and process design and included significant research papers from key contributors to platform product development to each of these topics. The product definition is mostly concerned with mapping the varied customer needs to the functional requirements intended for platform planning, product family positioning and platform selection. Most of these planning tools are basically static and they do not include the dynamic nature of product development where the initial plans have to be adapted as product designs evolve and the reality of engineering changes affect the project. The product design portion is mostly concerned with detailing optimization methods to design the correct products based on the functional requirements and frameworks to balance the need of distinctiveness with product commonality. While the product design methods provide useful tools for understanding product platforms design, they are centered on product parameter decisions and usually do not consider the effect to the product development project and balancing cost, scope and schedule on each of these decisions.

2.1.2 Product Commonality and Product Variety

Product commonality can be understood as a relevant measure of the design efficiency of a product platform. Higher commonality is usually related to a better design execution and is historically recognized as an enabler to achieve economies of scale as in words of Henry Ford (1923) referring to the Model T: "Any customer can have a car painted any color that he wants as long as it is black". Product commonality is usually included in the context of a product platform, but they are not synonyms: commonality refers to the sharing of components, processes or interfaces between different products while the platform is defined as the core technologies of a product, which are mostly shared. Therefore, commonality is related to design reuse while product platform is related to product architecture.

In his PhD dissertation, Boas (2008) starts with the concept of commonality or component-sharing as one of the potential tools to increase corporate profitability, however, he states that only few studies have emphasized the benefits and penalties of commonality. He further explores commonality in the context of complex product families in a set of seven case studies to bridge the current research with industrial practice and found two other phenomena not addressed in the literature: divergence –loss of commonality over time- and lifecycle offsets –the fact that complex product families are designed in a sequential manner-. His work is the most closely related to this thesis, as he is the only author to date that studied the dynamic aspect of product family development and acknowledges that product commonality decisions are made during the product development project that could worsen or benefit the profitability of the product family. Boas developed a simple cost model comparing the business case of developing a product family compared to independent product development for each product and found that in the context of divergence and lifecycle offsets, there is a breakpoint for deciding that a platform development could not be profitable as divergence and lifecycle offsets increase. The present work starts with Boas' framework of divergence and lifecycle offsets, but aims to have a deeper understanding on the effect of these phenomena on the execution of the product development project. Once the project has started, and a platform strategy has been already decided, the development team will not question the validity of the strategy, but will be more concerned on how to execute it within the desired scope, cost and time without changing the fundamental platform strategy.

A key issue with product commonality is the context-specific manner in which it is defined and assessed depending on the industry or technology. Understanding that sharing components is a key element of the platform strategy, Thevenot and Simpson (2006, 2007) and Alizon, Shooter and Simpson (2009) have developed different commonality indexes and compared them to other available indexes. The most simple commonality metric accounts for the ratio of common components to the total number of components; it can be computed for the overall product family or for each variant of the product family using the information from the bill of materials (BoM). More complex indexes have been developed to support decision making, and these metrics can include quantitative information such as the component costs and development costs as well as other qualitative information like shape, materials, manufacturing process or interfaces similarity. These metrics have been developed since many of the existing metrics were not useful because many of them were not easy to benchmark and assess relative to prior experiences; however, the latest metrics include an understandable context to support better commonality versus distinctiveness decisions.

As acknowledged by most of the authors of the product platform literature, the balance between distinctiveness and commonality is critical for successful product families. To create distinctiveness, product variety is another commonly addressed topic the product platform literature. Ramdas (2003) performed a comprehensive review on the existing literature and suggested a framework for addressing product variety decisions in the context of product development, marketing and operations. He uses the term "variegation" referring to the issue of distinct perception of one product to another in the same firm while differentiation is used in the context on how the product is different to the competition. In the context of this work on product platforms, we refer to differentiation similar to "variegation" that is creating distinctiveness among products in the product family. His framework addresses two different kinds of decisions: variety creation and variety implementation, the first one is more related to product development while variety implementation is more related to operations. In conjunction with the framework proposed by Boas, his framework is useful in understanding the tradeoffs of creating common or differentiated designs; however, the effects on project management are only tangentially addressed as in the case of Boas.

2.1.3 Project Management and Multi-Project Management

One of the objectives of this thesis is to link the project management practice into the current product development trend of developing product platforms / product families. The most common reference to the project management practice is the Project Management Book of Knowledge (2008), commonly identified as the PMBOK and published by the Project Management Institute (PMI). This guide provides useful guidance to the practitioners of project management through all its related activities, including the most common tools used by the industry. Similar literature on the tools and advice for the project management professionals include the AMA handbook of project management (2006) or the automotive industry specific guide for project management (AIAG, 1997). Nevertheless, these guides provide little guidance on how to execute more than one project at the same time, referred to as "program management" or "project portfolio management". Some of the specific multi-project management issues described in these guides include managing shared resources between projects, managing project's priorities, staggering projects and selecting the appropriate projects in the projects portfolio (AMA handbook of project management, 2006).

While all these guides for project management acknowledge the fact that project management has a continuous cycle of planning, execution and control, the presented tools for project control (like Gantt charts) are usually linear and not recursive (this will be further explained in section 2.1.4). Moreover, this recursive nature is also embedded in the product development process that is a well defined set of tasks to design and launch a new product, tasks that will be developed within a standalone product development project. The product platform product development is a specialization of the product development process, however, project management tools and guidelines for product platforms are not specifically addressed in these PMI and AMA guidelines.

"Thinking Beyond Lean" by Cusumano and Nobeoka (1998) is perhaps the best reference that bridges the gap of the execution of product platforms and concurrent development projects. They performed a study among several automotive projects in the mid 1990s and they concluded that the best companies organized their projects as part of a portfolio of projects instead of executing each project in isolation. They present the concept of multi-project management as a "set of conscious, planned efforts to link a set of projects strategically, through product portfolio planning, technologically, through the design of common core components, and organizationally, through overlapping the responsibilities and work of project managers and individual engineers".

They argue that the execution of product platforms (and specifically, automotive platforms) goes beyond product architecture and requires multi-project management as an organizational enabler to successfully execute product platforms. After studying four different design strategies for developing a product platform varying from a new design from scratch to concurrent technology transfer, they conclude that the later provides the advantages of savings in development costs, reduced lead time and growth in sales and market share. These advantages are achieved because the teams develop all the various product variants concurrently, allowing the company to diffuse new technologies among several platform variants faster and to simplify the development work by allowing mutual adjustment between products, task sharing and joint designs. In addition, they present also a discussion on the preferred organizational structure to achieve the concurrent technology transfer strategy and conclude that communication, coordination and integration must be considered both across functions and across projects simultaneously.

Their work acknowledges that product platform project execution is complicated and requires complex organizational schemes to develop it. They highlight the dynamic nature of product development by acknowledging that concurrent designs allow mutual adjustments for the design of various products by executing them concurrently, but they fail to discuss the implications of these mutual adjustments in the context of the development projects. They also do not discuss what happens to the product family as a whole as their study was more focused on individual projects, and what are the best management practices to achieve both product differentiation and commonization at the same time beyond the organizational structure of the project.

An additional resource on multi-project management is the case study of a second generation platform derivative in the automotive industry performed by Lenfle, Jouini and Derrosseaux (2007). Their study provides a good combination of the concept of multi-project management and the actual industrial management of the dynamic nature of product development in a platform driven organization. They found that development teams are more concerned with deciding what components to reuse instead of upfront product family design in order to converge the new product specifications with some of the existing components. They focus their research on the decisions made in the development process that ended with the product concept. They conclude that this convergence process is changing the traditional V model into a W model where the design team has to comply with two rules: satisfying new requirements and reusing previously designed components for a new application.

2.1.4 System Dynamics Applied to Project Management

Projects might evolve differently than planned and this may cause substantial schedule and cost overruns that will undoubtedly affect the profitability of the firm. A project will be understood as a finite collection of tasks, however, these tasks are usually understood as linear or static, but in reality projects are dynamic and exhibit varying behavior over time (de Weck, Lyneis, 2009). This dynamic behavior is driven by the embedded feedback cycle in project management: the planning, execution and control cycle: as a project is executed, a deviation from the plan may be found and a control action is required to update the plan to get the project back on track. As acknowledged by de Weck and Lyneis, many of the project management techniques like the Program Evaluation and Review Technique (PERT), Earned Value Management (EVM) and the Critical Path method do not capture the dynamic behavior of projects, so it makes project management a particularly good application for System Dynamics.

System dynamics is a computer simulation methodology used to enhance learning in complex systems, based on the theory of nonlinear dynamics and feedback control originally developed by Jay W. Forrester at MIT (Sterman, 2000). Lyneis and Ford (2007) performed an integrative survey for the application of system dynamics to project management and concluded that it has been one of the most successful fields for the application of the system dynamics methodology in both academic research and real-world applications, even creating the term "project dynamics" to refer to this trend. They categorized four different groups of application based on the central concept of the model in four structure groups: project features, rework cycle, project control, and ripple and knock-on effects. These models have captured many of the effects of project management and the effect of the feedback loops on productivity, resources, staffing, management pressure, meeting project's performance targets, etc in a wide range of industries. Most of the models can be categorized in any of the four basic structures described and there is little variation driven by the nature of the projects (construction, product development, software development, etc). Moreover, they also have categorized the objective of these models in four areas: post-mortem assessment for disputes and learning; project estimating and risk assessment; change management, risk management and project control; and management training and education. However, Lyneis and Ford acknowledge that the spread of system dynamics in project management is much less than traditional tools like the critical path method since it has a more strategic / tactical nature and the relatively complex nature of the tool compared to traditional tools.

In a good example of the application of system dynamics to the management of complex projects, Lyneis, Cooper and Els (2001) present a single case study of a successful use of system dynamics in the Peace Shield Air Defense project applying management lessons from a system dynamics model. This model was an extension from prior experience of using system dynamics model that supported the development of a bid for a troublesome project. While several examples of the application of system dynamics may be found in the literature (Lyneis, Ford, 2007), this paper provides a good description of the heart of project dynamics, the rework cycle which includes work accomplishment, feedback effects and knock on effects. The basic idea on the rework cycle is that a fraction of the tasks are finished and the others create rework that is usually not accounted for in traditional project management tools, this rework is sometimes found years after the work was originally done. However, one of the limitations is the single case study approach that may not be applicable for commercial product development projects.

While Lyneis and Ford (2007) provide an overall picture of the application of system dynamics to projects, platform development is more concerned with product development projects. In a more focused application of "project dynamics", Ford and Sterman (1998) developed a system dynamics framework to model the product development process. The process is based on a multi-phase structure (i.e. product definition, design, prototype testing, etc) with gateway approvals and check points to uncouple the dynamics of product development projects from the common dynamics of projects and the rework loop (effects on resources, scope, targets, etc). Their structure acknowledges a highly interdependent process in which iteration is very important. In their work, they provide the stocks and flows and feedback structures providing very useful guidance on the modeling of product development processes. One of their key contributions is the complete definition of their model, which is not commonly available (Lyneis, Ford, 2007). This framework can be further applied to model the differences of the single product development process when it is applied to product family development.

Finally, the above literature deals with system dynamics applied to single projects. To address the issue of resource allocation in multiple projects, Repenning (2000) developed the only found example of the application of project dynamics into a multiple projects approach despite the general trend of companies focusing on a portfolio of projects instead of single projects in isolation. While they provide a good insight into the resource allocation issue and how the allocation of resources to the early phases of product development can solve the numerous issues found downstream, their model does not include the basic rework model with quality and productivity loops commonly addressed in the system dynamics models and there is no mention of commonality amongst projects. The basic idea of the system dynamics model to be developed in this work is to understand the rework loop existing in a multi-project management setting on a set of projects interconnected by a common product platform project. In such a situation rework is not only created within each individual project but rework is also created by adjacent projects.

2.1.5. Automotive Industry Platforms Related Literature / Platform Design Process

Given that the basic case study for application of the system dynamics methodology is an automotive platform development, an additional stream of bibliographical research was performed on the specific knowledge created for platform development in the automotive industry to assess some of the typical behavior of automotive platform development projects. The previously cited work by Cusumano and Nobeoka (1998) is perhaps the best example of bridging project behavior into platform product development. A common element in this literature stream is the definition of the automotive platform based on the vehicle architecture and by "platform" they generally refer to the chassis and underbody systems, sometimes broadened by the addition of the engine and drivetrain systems as well.

Muffatto and Roveda (2000) provide one of the best intents to bridge the platform concept acknowledging the differences depending on the focus of the author; some of the narrowest definitions of a platform came from the automotive industry cases referring to the vehicle architecture concept of a platform described previously. In this work, they analyzed the platform development process in three different industries, the automotive industry being one of them. They started using a platform concept aligned with Meyer and Lenherd (1997) and developed a framework to analyze the three platform development process case studies; this framework includes the interconnection between the strategy (product family), the product structure (product architecture), the technology (product and process) and the organization (product development process and organization itself). The product development project effects are analyzed in the organizational piece of their framework and it was found that the organizational setting and the resources mobility may be as important as the flexibility of the product architecture in platform product development.

In prior work Muffatto (1999) performed a more focused case study of five major Japanese automotive companies to understand their platform strategies, recognizing that the platform approach is simultaneously a technical, strategic and organizational issue. This is the best example of the description of the platform approach in the automotive industry which details some unique elements in the context of a motor vehicle. One of the key observations is the separation of platform (underbody) development from vehicle (aesthetic or upperbody) development. In this way companies can create new looks for their vehicle with a shorter lead time since the platform development is very time consuming. This specific characteristic can also impact the organizational structure and relationship between the platform and vehicle development. Another key observation in the product development process is the derivation of new platforms from prior models, and then the derivation of new models from these platforms. The main contribution of this paper into the project management context is the acknowledgement that platform development modifies the product development process and also the organization; however, it does not correlate this effect with project outcomes beyond the acknowledgement of a reduced development time nor does it explore the tradeoffs beween the project's costs, scope and risks as the project progresses.

Mahmound-Jouini and Lenfle (2010) performed a case study on a mass production automotive company to understand the context of reusing designs from a prior automotive platform to a second generation set of products to be developed from the same platform. Their research fills an important gap in the literature on the analysis of the platform design process itself including the evolution to future products and the interplay between the platform and the products reusing it. They focused on the design decisions throughout different design phases during the platform renewal project and the underlying knowledge required to support those decisions, that include the product development knowledge: market, technical and economical knowledge plus a fourth new type of knowledge: platform knowledge. In their research, given the four types of knowledge, they create two design rules during platform development: reusing existing components to improve commonality and "lean design" to optimize a base platform that will be scaled up to avoid overdesign. They found that during the design process, these rules are questioned and the resulting platform design may convey exceptions to these rules in what they call "smart reuse" that means deciding the best available option given the existing trade-offs. They conclude that the commonality and differentiation plan (Robertson, Ulrich, 1998) can be used as a framework not only during the planning portion of the development but also throughout all the platform lifecycle. While they acknowledge the difficulties and the dynamic nature of platform development, they fail to describe what the impact is for each of the derivatives whenever a design decision is made and what the projects impacts are on: cost, scope and schedule. Nevertheless, they also support the fact that it requires a suitable organization for these decisions, specifically, the role of the "platform director" and the "core platform manager".

In addition, a separate literature stream aims to optimize product's design in the context of product platforms and product commonality. In the most relevant example of these methods applied to the automotive industry, Suh, de Weck and Chang (2007) developed a methodology based on flexible designs and future product uncertainty to identify the potential body in white components in a vehicle platform that should be designed to be flexible and easily modified for future product generations or derivatives for overall lower product costs. While the present work aims to understand the effect of product commonality and product platform strategies to the development project(s); the nature of the product and its architecture needs also to be acknowledged. Nevertheless, it is not the intent of this thesis to address the technical engineering challenge of designing complex vehicles which have to achieve several and more stringent requirements, tighter development schedules and lower costs pressure. Other relevant product optimization techniques can be found in Simpson's compilation on product families design (2006).

2.1.6: Literature Research Summary

Several other topics appear in the product platform literature that also contribute to the platform body of knowledge, however while this topics can also contribute to our understanding of project management for product platforms, they are not part of the main context of the cost, scope and schedule tradeoff that are more related to the development project than other topics that may be related to the product itself, examples of these topics include:

- Product architecture (modular or integral architectures)
- Mass customization
- Frameworks for product and platform performance optimization
- Platform product development process (top down vs bottom up)

The main literature research was divided into the five topics detailed above as follows:

- Literature Context: Product Platform Development
- Key area of research 1: product commonality and product variety
- Key area of research 2: project Management
- Selected analysis tool: system dynamics
- Case study application: automotive Industry

To understand the contribution to each of the five topics above, table 2.1 provides a summary of the literature research and summarizes the literature gap. In the left column, the five main literature research branches are subdivided into the topics that will be addressed in the present thesis, each row represents a topic. The upper rows divide these five main topics into a set of different literature sources, where each column represents a different stream of work. The matrix represents the intersection of the topics being addressed by each of the literature sources, where the black cells represent a main literature source for the topic and the grey cells represent a topic that was addressed by the author, but is not part of the main contribution of their work.

The first conclusion from the table summary above is that the system dynamics technique has never been used to model the platform product development approach; this technique and tools have only been used in one example of a multi project management context. This is the key literature gap that is intended to be bridged with the current thesis. Moreover, the current thesis can be thought of as a continuation of Boas' research about commonality in complex product families, where divergence and lifecycle offsets are present; however, the main approach is not to understand the tradeoff of developing a product family instead of developing each product in isolation, but understanding how to better plan and adapt the corresponding product development projects by making the cost, scope and schedule tradeoffs once a product platform approach has been selected and it has to be executed.

2.2 Proposed Framework

Most of the literature research recognized that product platforms are a useful approach to drive improvements in product lead time, cost reduction and quality improvements. One of the most important aspects for a successful product platform is the upfront planning of the product family and the underlying product platform (Meyer and Utterback, 1992; Meyer and Lenherd, 1997; Ulrich and Robertson, 1998); however, the execution of the plan itself can become as relevant as product planning.

			Meyer and Utterback (1992)	Meyer and Lenherd (1997)	Ulrich and Robertson (1998)	Simpson, Siddique and Jiao (2006)	Boas (1998)	Thevenot and Simpson (2006, 2007)	Alizon, Shooter and Simpson (2009)	Ramdas (2003)	PMBOK (2008)	AMA handbook of PM (2006)	Cusumano and Nobeoka (1998)	Lenfle, Jouini, Derrosseaux (2007)	de Weck, Lyneis (2009)	Lyneis and Ford (2007)	Lyneis, Cooper and Els (2001)	Sterman and Ford (1998)	Reppening (2000)	Muffatto and Roveda (2000)	Muffatto (1999)	Mahmoud-Jouini and Lenfle (2010)
Context: Product Platforms and Product Families Product Development		Product Platform Definition	■	■	▨	▨	▨	▨	▨				▨	▨						■	■	
		Product Families / Platforms Planning and Strategy	■	■	■	■	▨			▨			▨	▨						▨	▨	
		Product Families / Platforms Execution				▨	■						■	■						▨	▨	■
		Platform process benefits	▨	▨	▨	■	▨	▨	▨				▨	▨						▨	▨	▨
		Platform process tradeoffs	▨	▨	▨	■	■	▨	▨				▨	▨								▨
Key area of research within the Platform Approach: Commonality and Variety		Product commonality	▨	▨	▨	■	▨	■	■	■										▨	▨	▨
		Product variety	▨	▨	▨	■	■			■										▨	▨	▨
		Commonality Measurement			▨	■	▨	■	■	■												
		Divergence					■							▨								
		Tradeoff between commonality and variety	▨	▨	■	■	▨	■	■	■			▨	▨						▨	▨	■
Key area of research within the Platform approach context: Project Management	Single Project	Methods, processes and tools									■	■			▨	▨	▨	▨				
		Scope, cost, schedule and risk management									■	■			▨	▨	▨	▨				
		Team organization and communication									■	■			▨	▨	▨	▨				
	Multiple Projects	Definition of multi-project management									■	▨	■		▨							
		Methods, processes and tools									▨	▨	▨		▨				▨			
		Scope, cost, schedule and risk management		▨			▨				▨	■	■	■	▨				▨			
		Schedule management: lifecycle offsets					■						■	▨					▨			
		Team organization and communication		▨							▨	■	■	▨	▨				▨			
		Shared resources management and team organization	▨							▨		■	■	▨	▨				■			
Analysis Tool: System Dynamics		System dynamics application to project Management													■	■	▨	▨	▨			
		Rework Cycle, feedback effects and knock on effects													▨	■	■	▨				
		Product Development multiple phases models														▨		■				
		Multi projects simulations														▨			■			
Case Studies Application: Automotive industry		Definition of automotive platform				▨	▨						▨							■	▨	▨
		Platform development and vehicle development differentiation					▨						▨	▨						▨	■	
		Case studies of platform development in the automotive			▨	■	■						■	■						■	■	■

Table 2.1: Literature research summary and literature gaps.

Commonality is one of the key aspects of a product platform to achieve the expected benefits (Thevenot and Simpson, 2006, 2007; Alizon, Shooter and Simpson, 2009), however, it has to be balanced with product variety (Ramdas, 2003) to ensure the product's success in the market (Ulrich and Robertson, 1998). Nonetheless, product commonality and differentiation has been usually addressed in the context of the static product planning, however, as the product is being developed, "divergence" appears and the expected commonality is reduced compared to the original plan (Boas, 2008).

All the products that are part of the product family will be developed in a product development project, and the execution of the product family requires multi-project management knowledge and techniques to fully achieve the benefits of the product platform (Cusumano and Nobeoka, 1998). The main sources of project management knowledge deals with methods, applications and tools to handle the project's scope, cost, schedule and risk (PMBOK, 2008; AMA handbook of Project Management, 2006), while all these are still applicable for a multi project management setup, few tools and knowledge have been developed for managing multiple projects concurrently, where specific issues appear in a multiple projects environment: having different project schedules, known as lifecycle offsets (Boas, 2008; Lenfle, Jouini, Derrosseaux, 1997), sharing project's resources or determining priority between all projects (AMA handbook, 2006).

As previously stated, the product platforms and product commonality literature is more concerned with the original product planning, however, little emphasis is put on project execution. On the other hand, most of the project management literature acknowledges that projects are dynamic entities driven by the project control cycle, however, most of the project management tools do not address this dynamic nature (de Weck, Lyneis, 2010). Due to the dynamic behavior of project management, system dynamics has been a very successful tool for understanding project management (Lyneis and Ford, 2007) of individual projects. Most of the simulation models are built around the rework cycle (Lyneis, Cooper and Els, 2001), and many features have been added to this basic model like the multi-phase nature of the product development process (Ford and Sterman, 1998), nevertheless, only one model was found that addresses the resource allocation in a multi project management setting (Reppening, 2000). The intent of this thesis is to develop a system dynamics model for a multi project management setting that specifically addresses the rework created by divergence and lifecycle offsets in a product platform development project with important considerations of commonality in the automotive industry.

The automotive industry has several examples and years of industrial application of the platform approach that has been refined over the years. While most of the product platforms and commonality knowledge is valid for many physical products, the automotive industry provides a slightly different approach to the term "platform". Different to other industries, this term is common knowledge in the automotive context (Stewart, in Popular Mechanics, 2008) and is usually used to refer to the underbody systems of a vehicle that are usually shared between brands and sometimes with other OEMs. The literature has tried to standardize the platform concept (Muffatto and Roveda, 2000). Besides the definition of the platform itself, the design and development of the product family has another key difference where the product can be thought of as the sum of two major subsystems: the underbody or "platform" and the upperbody or "Top Hat" (del Puerto Valdez, 2010). This architectural setup allows OEMs to design the underbody and the upperbody systems independently, and it also allows a different organizational structure adapted to the specific development process for the platform and the vehicle or Top Hat (Muffatto, 1999). Given the distinct approach in the automotive industry context, additional case studies (Mahmoud-Jouini and Lenfle, 2010; Cusumano and Nobeoka, 1998; Muffatto, 1999) were added to the literature research to validate the findings from the case study in similar industry conditions.

3. Case Study

3.1. Product Family Overview

The case study is based on the design and development of an automotive platform from its project start to the design completion of the complete first set of derivatives, representing approximately two years of design and development of a new vehicle project. The overall product family had a global scope and comprised the execution of 18 platform derivatives whose details are summarized in table 3.1; the information has been disguised for confidentiality purposes. As seen in the table, from the 18 different derivatives, there are 9 different products or body style executions. The real project names have been disguised with a different canine to emphasize that all these products are based from the same product platform. The products will be assembled in 7 different manufacturing plants and will serve four major regional markets. To further increase the complexity of the platform scope, the 18 derivatives are offered with various engine options for a total of 16 different engines across the platform. Each of these engine options were managed within each of their applicable products / projects.

The product family was divided into three "waves", and each wave had a different design and development timing where the wave number represented the first and last group of derivatives to be launched into the market, but also represented a different vehicle segment. In terms of size and forecasted sales volume, the first wave was the smallest and lightest vehicle from the platform, but represented the highest sales volume; the second wave and the third wave followed. To simplify the context of these vehicles, the names "Dog", "Wolf" and "Coyote" are a fair representation of the size and weight of each of the waves so it will be called the "canine platform" or "canine product family" to the set of all the different derivatives. A specific discussion on the development timing is included in section 3.2.

The canine platform was developed as a major update from a previous platform which originally served the dog-like and wolf-like vehicle segments in one of the regions. The renewed platform was planned to transition from a regional scope to a global scope and represented the convergence of five legacy platforms into one platform, following the industry trend to reduce the number of platforms and therefore increase the number of vehicles per platform (A.T. Kearney, 2010).

The specific effort to commonize several vehicles into a common platform or architecture translates to the relevance of these products to the company under study. The significance of the canine platform in terms of global vehicle sales in the company was intended to increase from 4% to approximately 19%; the right execution of the projects and the related products had a strategic relevance to the company therefore received a preferred involvement of the senior engineering leadership team. As quoted inside the company, "(the "canine" platform) is the most important project for the company".

The engineering department was already familiar with developing product platforms and they were aware of the benefits of commonality; however, their prior experience was limited to developing products for a specific market (region) and it was the first time these vehicles in these specific segments were developed using a single platform with a global scope. The company has already successfully developed other global platforms, nevertheless, they were developed by a different engineering team in another region and this is the first time this specific regional engineering department was leading the design and development for a market other than their own market and their regional assembly plants.

No	Product (Body Style)	Region	Plants	Wave	Launch Order	Volume relevance	Platform Project Manager	Top Hat Project Manager	Project (Finance)	Project volume relevance
1	Shepherd	Egypt	Alexandria	1 - Dog	1	1	A	B	Egyptian Dog	1
2	Shepherd	Rome	Alexandria	1 - Dog	1	16	A	B	Egyptian Dog	
3	Retriever	Egypt	Alexandria	1 - Dog	1	7	A	C	Retriever	7
4	Shepherd	Greece	Athens	1 - Dog	2	12	A	B	Greek Dog	2
5	Shepherd	Greece	Troy	1 - Dog	3	13	A	B	Greek Dog	
6	Hound	Greece	Athens	1 - Dog	2	6	A	B	Greek Dog	
7	Hound	Greece	Athens	1 - Dog	2	17	A	B	Greek Dog	
8	Terrier	Greece	Athens	1 - Dog	2	8	A	B	Greek Dog	
9	Terrier	Greece	Athens	1 - Dog	2	18	A	B	Greek Dog	
11	Shepherd	Maya	Tulum	1 - Dog	4	4	A	B	Mayan Dog	5
10	Wolf	Greece	Athens	2 - Wolf	5	9	A	D	Greek Wolf	8
12	Steppe Wolf	Greece	Athens	2 - Wolf	5	15	A	D	Steppe Wolf	12
13	Wolf	Egypt	Memphis	2 - Wolf	6	2	A	D	Egyptian Wolf	3
14	Tundra Wolf	Egypt	Memphis	2 - Wolf	6	10	A	E	Tundra Wolf	9
15	Wolf	Maya	Tulum	2 - Wolf	7	14	A	D	MayanWolf	11
16	Coyote	Maya	Chichen Itza	3 - Coyote	8	5	A	F	Mayan Coyote	6
17	Coyote	Egypt	Thebes	3 - Coyote	9	3	A	F	Egyptian Coyote	4
18	Jackal	Egypt	Thebes	3 - Coyote	9	11	A	G	Jackal	10
	Total = 9	**Total = 4**	**Total = 7**	**Total = 3**	**Total = 9**		**Total = 1**	**Total = 6**	**Total = 12**	

Table 3.1: Case study product family details

Besides the evolution of the engineering department towards global product development, the platform and the products themselves are also required to be significantly updated in order to be competitive in the marketplace worldwide and to comply with the latest industry regulations applicable to their respective regions. Along with the new underbody development, a new upperbody with new styling for all the planned derivatives was also included in the scope of changes for each wave. The main reasons that drove the strategic changes to the products were:

- Increased platform scope from regional to global vehicles and therefore compliance to global standards, which was commonly the highest among the regional standards
- More stringent safety regulations
- Support new exterior and interior styling to differentiate between platform derivatives with competitive vehicle roominess
- Support a more aggressive effort to differentiate the base and the premium vehicles
- Added capability for larger engines not existing in the previous platform
- Increased wheelbase
- Improved fuel economy
- Vehicle weight reduction
- Support intra-company design optimization and sourcing strategies
- New emissions standards

As expected from the extensive list of change drivers above, there was an evident conflict for the engineering team. The product had to be significantly updated; however, these updates should be accommodated into an already existing platform with a predefined range of performance, also known as platform extent (Seepersad et. al, 2000, de Weck 2005). Before the project starts, a significant effort of product planning and vehicle architecture studies were performed to determine the most effective way to accommodate the new requirements into the carryover architecture. The result of these efforts was the commonality and differentiation plans that were used as input for the development team to assess and execute specified changes to the vehicle. Obviously, the related project profitability studies were performed based on these changes in order to define technical and financial feasibility of the project. These studies included a significant investment related to updating the existing assembly plants to accommodate the new vehicle architecture and the bill of process dictated by the vehicle architecture.

The sum of all these factors: significant vehicle changes in carryover architecture, the new global scope and the increased number of affected manufacturing plants, required also a different strategy to coordinate the overall product family scope. First, the role of the platform project manager was created (similar to the other global platforms) to further emphasize the relevance of cross-vehicle sharing and to drive an efficient design and development of the underbody. As a counterpart, a set of additional project managers in charge of the exterior and interior styling sere appointed, they were called the "Top Hat" project managers. In addition, the planned 18 derivatives (see table 3.1 for further references) were organized into 12 projects to allow a manageable scope. The selection of the projects was based on how their financial information was managed. Generally speaking, a project was a combination of a product and a manufacturing plan. All the projects, including the "platform project," were coordinated by a product family director, to which all project managers reported.

The following sections are intended to further explain the details on how the projects were organized and how they progressed during their design and development. The case study details only two years of the overall project timeline, and its difficult to determine if all these projects were successful based on their overall product market success. Nevertheless, based on preliminary testing, market research, extensive benchmarking and planned performance, all the products should be very successful in the market as they have already been successful in their prior versions. All the products are expecting an increased sales volume to what their predecessors had. In terms of the successful execution of the project's cost, scope and time, the project has been internally acknowledged as a successful project based on the internal project performance benchmarking and its high complexity and global scope.

3.2. Product Family Projects and Development Timing

3.2.1. Description of the Lifecycle Offsets and Overall Development Process

The product family under study was developed in a staggered manner, using the different waves as the preferred characteristic to group projects. As explained in the overview, each wave is a different vehicle segment and its derivatives are intended to cover the premium and non premium markets in different regions. The main driver for staggering the product launches was for ensuring a continuous stream of product launches within the company. Also the engineering expenses are spread out across several projects avoiding a significant engineering headcount increase for the company.

The product family high-level development plan is detailed in figure 3.1. This figure details each of the different projects and the main phases of product development: concept development, detailed design, testing and refinement, and production launch. The first key feature of the development of the platform was that the upperbody and the underbody design streams were separated as they could be developed relatively independent of each other (Muffatto, 1999).

As seen in the figure, only during the pre-planning portion of the product family (in gray) the three waves were assessed and designed concurrently. During this phase, most of the effort was expended on the first wave with a high level assessment of feasibility of the other two waves. After the first wave finalized their planning phase, they immediately started their concept development; however, the other two waves continued their pre-project phase until they started their concept development phase. The white space, with no work apparently being done on the second and third waves, represents the lifecycle offset between waves.

The first phase of each wave represents a conceptual design phase for both the upperbody and the underbody components. The development engineers develop these conceptual designs, and the integration team assesses these designs and overall product compatibility. As the team finds technical or financial incompatibilities, they update the conceptual design until the product and the project achieve compatibility. After all the conceptual designs for each of the major vehicle systems are selected, then the team continues to find a suitable supply base for its future production.

After the conceptual design phase, the underbody team continues to the detailed design phase, while the upperbody team continues assessing different styling alternatives. The detailed design phase is considered to be completed when all the underbody components achieve 100% geometrical compatibility for the vehicle and they have found a production source for all. During this phase, the upperbody detailed design starts when the team has selected a final styling theme; this overlap allows the required upperbody and underbody coordination to achieve overall product compatibility.

The next phase is the construction of the underbody prototypes and its respective testing and validation; these prototypes are built with production intent underbody, but with a surrogate upperbody. During the underbody validation phase, the upperbody continues with their detailed design that ends when the upperbody achieves 100% geometrical compatibility. When the upperbody and the underbody both complete each of these phases, the product goes over a major product and project review to authorize the release of the designed parts for production and the production tools development. This represents the headcount peak point for the project.

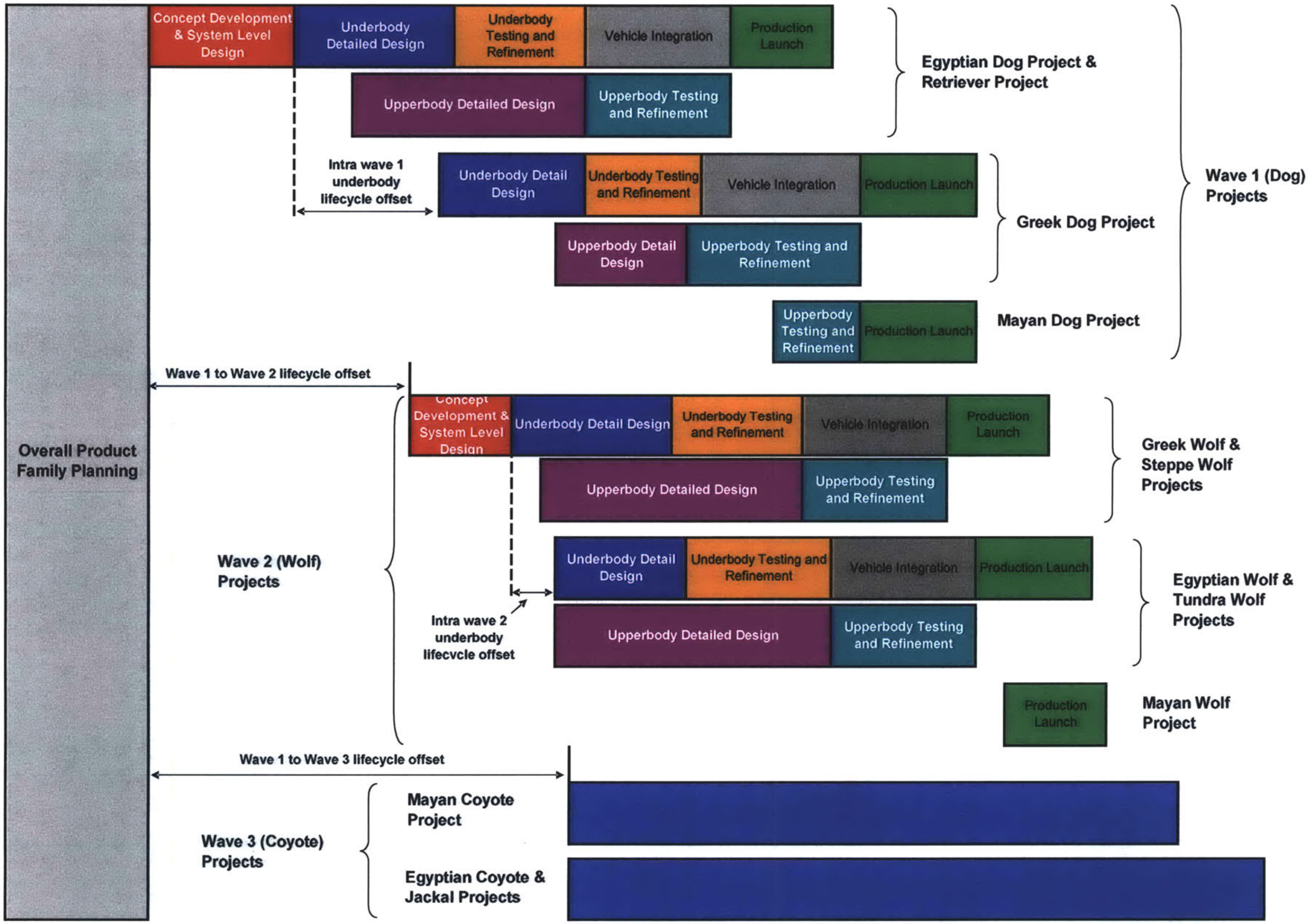


Figure 3.1: Product Family Projects development timing per project and per Wave

If the project is authorized, then the production intent prototypes (upperbody prototypes) are built for validation and refinement. During this phase, the underbody development is almost finished and has a support role for the overall vehicle validation and compatibility with the upperbody design. This represents a second opportunity for further refinement of the underbody components, although the anticipated changes are expected to be minor updates to the already validated design. After the vehicle is completely tested and validated, then it is authorized for its production launch phase.

The final phase of the projects is the production launch which is concurrent for both the upperbody and the underbody. This phase is more relevant to the manufacturing activities compared to the design and development activities, as they should be mostly finished after the validation phase. During the launch phase, production tools are calibrated to the design intent and the first production prototypes are built. This final refinement phase ends with the start of production of the vehicle.

This same process was applicable for the second wave and third wave, whose detailed plans were not available at the time this thesis was finalized. The blue bar represents the expected development time for each of the third wave derivatives per the overall product family plan. A key feature of the product platform approach is that the development time for the second and third waves was expected to be shorter than the first set of derivatives. Usually the first product pays the burden of the development since they have to ensure that the components can be reused for the future applications.

Finally, as seen in figure 3.1, another key feature of the project under study is that in both the first and the second waves two different upperbody and underbody workstreams were part of their development. This was driven because of having different engines and transmissions for each of the regions, along with different styling versions that were required in the other projects. While this is not efficient from a design and coordination perspective, it allowed the platform to have higher flexibility to accommodate all the regional requirements. These differences and reasons for staggered timing among waves will be further explained in the next section. These lifecycle offsets are also noted in figure 3.1, however, their length is much shorter than the lifecycle offset between waves.

3.2.2. Chronological Description of the Development Projects and Products

The easiest way to understand the "Product Family Project" is by describing the individual projects that were outlined in table 3.1 and figure 3.1, which are recommended to the reader to keep handy for the rest of the case study as they summarize the key similarities and differences between products. Most of the comparisons between projects will be done to the first platform derivative. The case study scope is bounded to represent only a portion of the product family project. Figure 3.2 details in light green the development phases and projects that were incorporated into the case study.

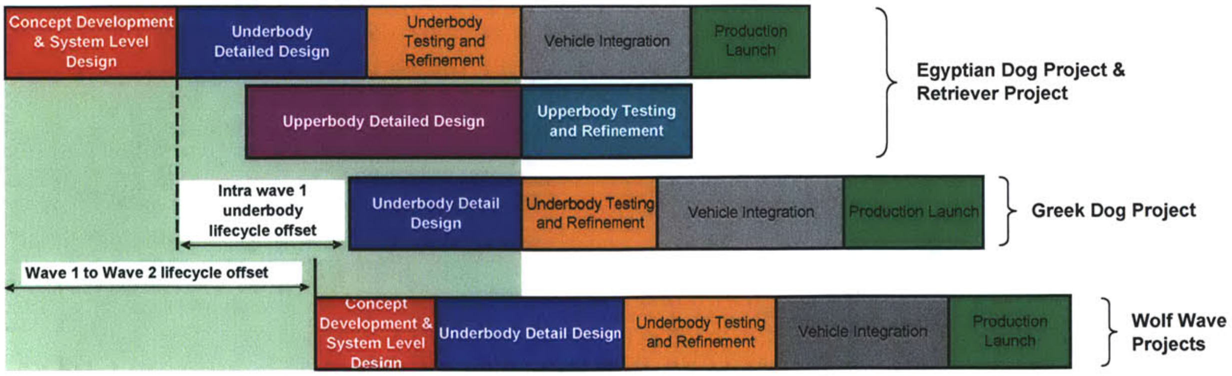


Figure 3.2. Scope of the case study.

The first project to be developed was named "Egyptian Dog" describing the region and the wave of the product. This will be called either the lead product or the lead project from now on. Besides being the first product to be launched, it was also the product with the highest sales volume and also shared its styling (similar to the "Shepherd" style) with all the other markets. Because of all these factors, the Egyptian Dog was used as a starting point for all the other products. The scope of the project included only one vehicle style, but offered with four different engines (none of them new to the company), manual and automatic transmission, and front and all wheel drive options. In terms of equipment, a low, medium and high series were also part of the scope of the project. Since a very similar product for the Roman market was required, with a much lower sales volume, it was decided to produce the "Roman Dog" in the same assembly plant in Alexandria and to be part of the same "Egyptian Dog" project.

Concurrent with the Egyptian Dog project, the "Retriever" had a completely different look compared to the Shepherd, and it was planned to be produced in the same manufacturing plant in Alexandria. The product would share the same platform and supply base with the Egyptian Shepherd product and was intended to cover the premium market in the same vehicle segment. Similar options in terms of powertrain and series differentiation were planned. Two engine options were available, one was shared with the Shepherd and the other engine was shared with other vehicles in the company. Besides the incremental work for a different engine and the new body style, some minor underbody changes were required to achieve premium segment performance, including larger brakes and different suspension tuning with the same architecture. The total incremental development required was significantly reduced compared to the first derivative and they were being developed concurrently to allow coordination and to understand the requirements of both vehicles.

The third project was named "Greek Dog", and it was the most complex project of all since it included the development of three new body styles (one shared with the first project), five total engines (two new engines, two existing engines from the prior vehicle and one shared engine with the lead product) and incremental features and options that were not part of the lead project development. The product would be assembled in the Athens' plant, and also in a sister plant in Troy that assembled the Shepherd-like product only. To avoid incremental investment, the two unique bodystyles assembled in Athens would be exported to other regions, nevertheless they were also part of the Greek Dog project. Because of the two new bodystyles development and the two new engines, the design workload was comparable to the lead project, even if the platform and most of the vehicle upperbody were shared with the first project. The project received a significant amount of management attention during its development, and required staggered upperbody and underbody design workstreams for their unique content as noted in figure 3.1. Since the assembly plant was located in a different region than the lead product's assembly plant, an alternate supply base was required for this project as well. To still obtain benefits from the common development, the purchasing and engineering teams looked for using global suppliers that could supply both manufacturing plants (Alexandria and Athens) using various manufacturing locations.

The last project of the first wave was the "Mayan Dog" which required an additional assembly location for the "Shepherd" bodystyle. The majority of the product design was shared with either the Egyptian or the Greek Shepherd and the vehicle design included the use of two common engines. Only slight modifications were required to comply with some local requirements meaning minor incremental new development was required for the Mayan Dog. While the design was predominantly common, the key challenge for the project was the distinct manufacturing footprint required for this product. Nevertheless, a significant portion of the suppliers, either from the lead or the Greek Dog project, were planning to supply the assembly plant in Babylon.

The second wave of vehicles were lead by the "Greek Wolf" project. The project was developed concurrently to the "Steppe Wolf" project, which required a slightly different styling for additional passengers capability. The platform required a stretch of vehicle capacity as these vehicles weighed more: new reinforcements were to comply with the safety and performance requirements for the vehicle segment. These new requirements, in addition to a new vehicle styling, were the drivers for most of the underbody changes. Yet, these projects had an advantage since the supply base was already developed as the product was planned to be assembled in the same Athens plant as the Greek Dog project. Also, this project required fewer engines than the first wave, many of them common to both waves. Due to the high similarity between the Greek Wolf and the Steppe Wolf, along with the shared assembly plant, these two projects were usually referred to as a single project.

The next two vehicles planned in the second wave were the "Egyptian Wolf" and the "Tundra Wolf". They were both assembled in the same assembly plant in Memphis, which brought an additional manufacturing location to the platform. In terms of design scope, the Egyptian Wolf initially planned to reuse the same bodystyle as the "Greek Wolf". The shared bodystyle of the Greek and the Egyptian Wolf changed as the design progressed, discussed in future sections. However, the "Tundra Wolf" had a completely distinct styling and was planned to cover the premium segment. This relationship is similar to what happened in the first wave with the Egyptian Dog and the Shepherd. Compared to the first wave, the projects had a reduced number of engines (3 in total). Yet, one of the shared engines represented a new engine development and brought a deferred underbody development phase as noted in figure 3.1. Due to the deferred start of production and a completely new exterior styling of the Tundra Wolf, a staggered upperbody design workstream was also required.

The last project of the second wave, the "Mayan Wolf" was comparable to its first wave counterpart, the Mayan Dog. The project also planned to reuse engines from other vehicles and the exterior styling of the Wolf product, being the third user of the same vehicle styling. The assembly plant was the same Tulum plant as in the first wave, so the selection of the supply base did not represent a significant challenge to the development team as it did for the first wave.

Finally, the third wave ("Coyote") was planned to be built in two new manufacturing locations. Two new exterior stylings were planned, one common between two regions and again a completely different styling for the premium market, the "Jackal." The third wave had similar weight and loads compared to the second wave, so the reuse of the front and rear suspensions and brakes was planned. Because the bodystyle was more similar to the first wave, some systems were to be reused, thus producing an efficient platform design that required less development. By the time this thesis was developed, the project was still under study and the conceptual design phase had not yet started. This wave considered to reuse engines from prior waves, however a new engine development was also considered in the project scope. Due to early phase of the third wave, it will not be as widely addressed in the case study as the other first two waves.

3.3. Project and Team Organization

3.3.1 Introduction to the Platform and Top Hat Structure

Platform product development and efficient platform design principles have been part of the company's product development system for several years. The company has continuously refined its platform planning and development processes to improve the performance of the products and projects. The most important change in the platform development process was the separation of the underbody and the upperbody development into two different coordinated design workstreams, which was already executed for the donor platform initial design. A relatively new change for the team was the separation of the product into "Platform" and "Top Hat" to refer to the underbody and upperbody of a vehicle. This could be considered the most basic decomposition of the product architecture.

The introduction of the Top Hat and Platform concept required a method to clearly understand what was included into the scope of each portion, so each part, product attribute and team member could be assigned into the appropriate portion of the product / project. In the context of the case study, the platform was considered to be all the vehicle parts and components that are not visible to the customer: i.e. braking system, exhaust system, engine, transmission, vehicle underbody structure, steering system, driveline, etc: the set of all subsystems that make the vehicle move, similar to a go-kart. Likewise, the differentiation of the vehicle is called the "Top Hat", which represents all the parts that the customer can see and touch. The Top Hat parts require extensive industrial design and styling, and have significant design churn during the vehicle development. As these parts make a derivative different to the others, the likeliness of sharing these parts with other vehicles is low and the parts are unique to each product. Figure 3.3 and table 3.2 further detail what was considered part of the platform and what was considered part of the Top Hat.

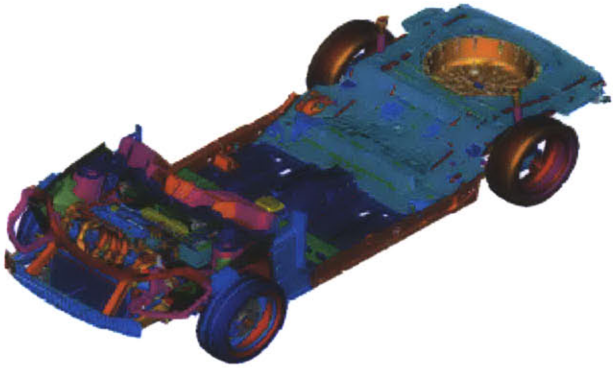


Figure 3.3: Product platform in the context of a vehicle – the "go kart'.

Platform (Underbody) Systems	Top Hat (Upperbody) Systems
Braking System	Exterior Stampings
Steering System	Exterior Paint
Front and Rear Suspension	Exterior Ornamentation
Front and Rear Floor	Front and Rear Fascias
Front Structure	Front and Rear Lamps
Electronic Modules	Exterior glass
Power Supply System	Exterior and Interior Mirrors
Wiring Harnesses	Moveable roof
Climate Control System	Mechanisms (Window, wiper, latches, etc)
Engine and accessory drive	Wheels
Transmission and shafts	Cluster and Passenger controls
Air Induction System	Entertainment and Navigation Systems
Exhaust System	Interior Trim
Cooling System	Instrument Panel
Fuel System	Air Bags and Steering Wheel
Tires (including spare and tools)	Seats

Table 3.2: Platform and Top Hat Systems Decomposition.

The product decomposition followed a relatively simple logic and it was easy to understand if a part or system belonged to the platform or Top Hat. Yet, there were some cases where the rule "if the customer can see or touch the part" fell into a grey area. For example, a section of a part was visible but it was a component of a subsystem that was mostly not visible, such as the exhaust tip. On the other hand there were not visible to the customer, but their design depended heavily on the styling of the vehicle, like a door latch. Nevertheless, the system-by-system convention was preferred to align with the existing organizational structure.

Another reason for the new organizational and project structure was to better guide the two different upperbody (Top Hat) and underbody (Platform) workstreams. As a result of the reduced number of platforms with the increase in derivatives from each platform, the proposed structure would balance the workload between project managers. The designation of the platform project manager and the recognition of the platform project were perhaps the most relevant changes compared to prior products. The following sections further details how the team and the project were divided and the effects were of this breakdown.

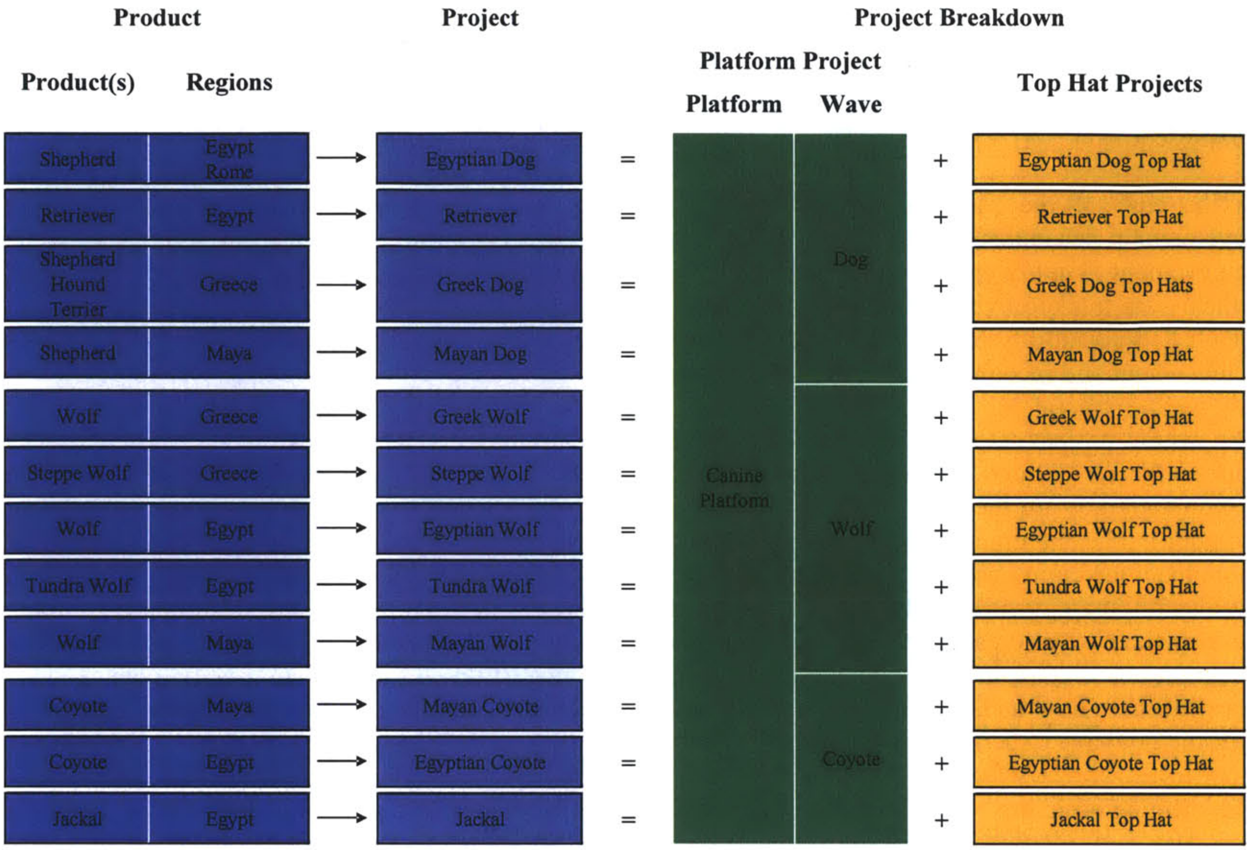
3.3.2 Platform and Top Hat Projects Roles and Responsibilities

The overall product family project decided to use the Top Hat and platform structure also to divide each derivative project into their respective platform and Top Hat projects. Using the agreed method above based in the vehicle's bill of materials (BoM), all the different parts were split and organized into their respective Top Hat or platform projects. Each derivative (based on the exterior styling) was considered a unique project itself, at least the upperbody portion. On the other hand, the platform portion of their derivative projects was handled via an overall platform project that included all the underbody systems for all the different derivatives from all the different waves. A similar structure was previously used in other product family projects with similar scope and this represented an effort to globally standardize the processes inside the company. The main objective for creating the platform project was to maintain product commonality by balancing decisions across multiple products and projects and to coordinate shared resources among all these products and projects.

Figure 3.4 below shows all the different projects for the case study: a total of 12 Top Hat projects and one platform project; in this case the platform project could also be further subdivided into the three waves, but we considered only one project for the company under study. The actual names of the projects (including the platform) were such that there was an intuitive coding that easily understood the nature of the region, the product, and the platform wave to further emphasize that they were part of the same product family and that there was commonality among them.

With all the parts and systems properly divided into Top Hat and platform, the different workstreams were divided as well. For most of the team members, they were not used to identifying if they were part of a Top Hat or a platform project, as they were previously members of one project with one or few products under its scope. This represented a significant change in the integration activities, which now had to differentiate between both the platform or Top Hat workstreams. All the different processes, tasks, meetings, approval cadence and costs were divided into the two workstreams and were almost identical. The main difference between both projects was on one hand the vehicle styling and appearance development processes that were not required by definition into the platform project. On the other hand, the platform project had a much larger scope based on all the different derivatives, but would strive to emphasize commonality among vehicles and would enable coordination between derivatives.

While the vehicle subsystems were relatively simple to classify into the Platform and Top Hat structure, the overall vehicle functional attributes were not, and this represented a challenge to coordinate the projects. These are all the different product characteristics related to performance attributed like vehicle safety, Noise Vibration and Harshness (NVH,) vehicle styling, vehicle dynamics, fuel economy, vehicle weight, overall vehicle quality and acceleration performance. All these attributes are achieved with all the different subsystems working together as a whole in the vehicle. The scope of the project includes delivering planned competitive functional attributes, one of the project's main deliverables.



Product			Project		Project Breakdown			
					Platform Project			
Product(s)	Regions				Platform	Wave		Top Hat Projects
Shepherd	Egypt Rome	→	Egyptian Dog	=	Canine Platform	Dog	+	Egyptian Dog Top Hat
Retriever	Egypt	→	Retriever	=			+	Retriever Top Hat
Shepherd Hound Terrier	Greece	→	Greek Dog	=			+	Greek Dog Top Hats
Shepherd	Maya	→	Mayan Dog	=			+	Mayan Dog Top Hat
Wolf	Greece	→	Greek Wolf	=		Wolf	+	Greek Wolf Top Hat
Steppe Wolf	Greece	→	Steppe Wolf	=			+	Steppe Wolf Top Hat
Wolf	Egypt	→	Egyptian Wolf	=			+	Egyptian Wolf Top Hat
Tundra Wolf	Egypt	→	Tundra Wolf	=			+	Tundra Wolf Top Hat
Wolf	Maya	→	Mayan Wolf	=			+	Mayan Wolf Top Hat
Coyote	Maya	→	Mayan Coyote	=		Coyote	+	Mayan Coyote Top Hat
Coyote	Egypt	→	Egyptian Coyote	=			+	Egyptian Coyote Top Hat
Jackal	Egypt	→	Jackal	=			+	Jackal Top Hat

Figure 3.4: Breakdown of the twelve derivative projects into one platform project and twelve Top Hat derivative projects.

The functional product performance attributes were lead by the vehicle integration workstream, which was concerned with the overall product performance regardless if they were part of the Top Hat or the platform. This workstream is in charge of developing the vehicle targets, assessing the overall attributes throughout the project, ensuring all the vehicle attributes are delivered based on the vehicle testing and identifying proposed corrective actions if the attribute assessment or test did not achieve the proposed target.

It could be argued that there are some functional performance attributes that could be divided into the Top Hat and platform structure. For example, braking performance is delivered mostly by the braking subsystem, however, a lighter vehicle can have a better performance. Another example is fuel economy that is predominantly driven by the engine size and the transmission ratios, however, the vehicle aerodynamics and the vehicle weight factors in as well. Instead of classifying these functional attributes in Top Hat and platform, they were selected to be delivered jointly by both projects, but lead by the vehicle integration workstream. The vehicle integration workstream had the expertise on how the different design variables mapped to the functional attributes and could recommend corrective actions (either for the Top Hat or the platform) that would be agreed upon by the platform or the Top Hat project managers respectively or jointly.

The scope of the project is bound by the delivery of the vehicle and its functional attributes, nevertheless, the two other main project deliverables are the project costs and schedule. In the case of project costs, it was fairly simple to classify the costs as Top Hat or platform based on the corresponding part that incurred these costs. The platform and the Top Hat projects tracked their financial performance through their respective finance team and they included the variable cost, the tooling and facilities costs, and the development costs. Furthermore, the Top Hat team was responsible for integrating the overall vehicle and projects costs and prepared the overall business case with its corresponding forecasted revenue and net present value (NPV). The underbody represented roughly 60% of the overall production cost.

In the case of the schedule, given that the platform team had several derivatives with different development schedules, it was also decided to have the Top Hat projects monitor the overall schedule achievement as an overall project. Nevertheless, some platform (underbody) milestones were required, which were tracked by the platform project itself. In general, the overall project schedule achievement was the responsibility of both projects, but it was the Top Hat project that traced it more closely for its achievement and recommended corrective actions when a schedule overrun happened or was forecasted.

The overall project activities or tasks were also divided into the Top Hat and platform. These activities can be further divided into other classifications: the design activities and integration activities. The design activities are the ones related to the development of each of the vehicle components (in relative isolation). The integration activities understood are the activities that put together the information for more than one component and put them together into a higher system level or a vehicle level. In general, the overall strategy of dividing the Top Hat and the platform projects did not affect the usual product development workstreams of the development engineers in their design activities; however, the integration activities, including manufacturing and marketing, suffered the most significant change since many of them now had to differentiate if the task was related to the platform or the Top Hat, an activity they did not have to perform in the past.

The meeting structure to monitor the progress of the project was also differentiated to follow the same Top Hat and platform structure. A similar meeting structure as used in the prior project was implemented: change control meetings, engineering attributes meetings, project progress review meetings, vehicle styling meetings, etc. All these meetings, but the ones related with the studio and styling design, were still in place; but their respective agendas now captured if the discussion was intended to review either a platform, a Top Hat or in other cases, a joint discussion about the overall project.

Similar to the vehicle styling for the tophat project, a unique deliverable for the platform was the overall underbody commonality. Even if commonality was expected from the upperbody systems with similar advantages, the underbody of the vehicle was expected to have a much higher degree of commonality and shared systems across different derivatives. The only other deliverable unique to the underbody or platform team was the development of an underbody prototype to validate most of the underbody systems. Following the principles of the vehicle architecture, any upperbody could be used to validate the underbody since it was intended to be a flexible architecture in order to accept different upper shells without any major performance deviation. The underbody prototypes allowed the platform team not only to validate their respective parts, but also the underbody as a whole.

The key roles and responsibilities of the Top Hat and platform projects are summarized in table 3.3. In summary, the project distinction of the Top Hat and platform workstreams represented a more complicated method of coordinating the projects, however, as the product families became more complex, coordinating these projects grew to be more of a challenge. The presence of the Top Hat and platform projects required the teams to differentiate the information (for Top Hat or platform) for later integration it to an overall project level. This can be inefficient, yet, it allowed each of the teams to focus on their key deliverables: commonization or differentiation. While the new structure represented a challenge for the integration activities, the design activities followed a very similar process, and overall the project achieved a similar coordination after the integration activities were learned and team members became familiar with the new structure.

Activities	Description	Platform Project	Tophat Project
Meetings	Underbody management reviews	Lead	Support
	Vehicle management reviews	Support	Lead
	Project progress Review	Shared [1]	Shared [1]
	Change Control Meetings	Shared [1]	Shared [1]
Prototypes	Complete verification and validation prototype	Support	Lead
	Underbody Prototype Build and Validation	Lead	N/A
Tasks and workstreams	Underbody parts drawings development and mockup	Lead	Support
	Upperbody parts drawings development and mockup	Support	Lead
	Vehicle integration workstreams (forecast and testing)	Shared	Shared
	Marketing workstreams	Support	Lead
	Assembly operations workstreams	Shared - Platform Only	Shared - Top Hat Only
	Sourcing workstreams	Shared - Platform Only	Shared - Top Hat Only
Product or Project Deliverables	Overall vehicle quality	Shared - Platform Only	Shared - Top Hat Only
	Vehicle functional attributes	Shared - Support	Lead - Overall integration
	Design commonality	Lead	N/A
	Vehicle Weight	Shared (Lead)	Shared (Support)
	Vehicle styling	N/A	Lead
	Project Schedule	Support	Lead
	Underbody specific workstream schedules	Lead	Support
	Upperbody specific workstream schedules	Support	Lead
	Projects costs	Shared	Shared
	Project business case	Costs only	Costs, revenue and NPV

1/ Note: The meeting agendas specified if the topic was intended to adress the platform, the tophat or both jointly.

Table 3.3: Platform and Top Hat projects roles and responsibilities.

3.3.3 Modified Matrix Report System

The engineering department has been historically organized in a conventional matrix report system where the engineers report dually to the project and to their functional departments. The conventional matrix system was modified to accommodate the new organizational structure as illustrated in figure 3.5. The key changes are the introduction of an incremental platform project manager for the platform project and also the distinction of the platform or Top Hat project as explained in the prior section. To identify all the team members of a vehicle project, two rows are now needed (one for the platform project and one for the Top Hat project).

The four functional engineering areas are further subdivided into other vehicle subsystems such as suspension, brakes or steering in the case of Chassis Engineering; or fascias, structures, closures, instrument panel or interior trim in the case of Body Engineering. The dual report is achieved with the engineers reporting directly to their functional area, as well as a dotted line report to the functional engineering manager representatives in the project. The engineering managers also a dotted report to the product family director with a direct report to their functional engineering directors which themselves have a direct report to the product development senior leadership.

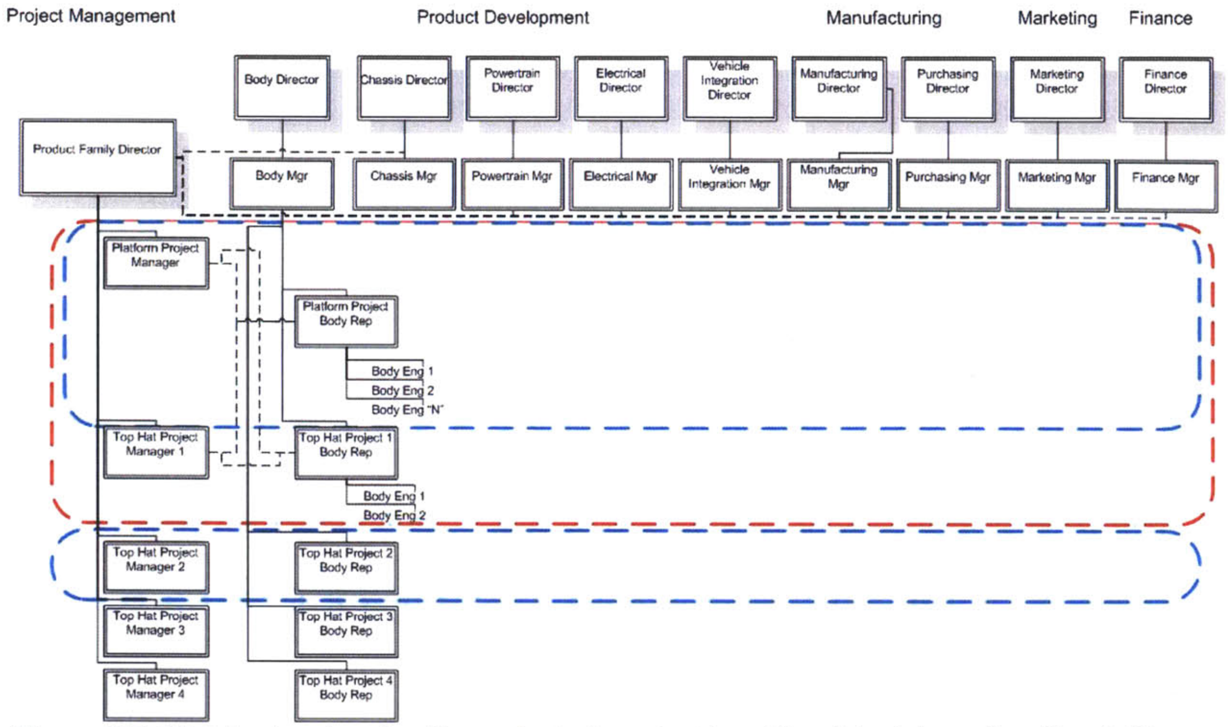


Figure 3.5: Matrix structure for the project. A project is achieved by integrating the platform and the Top Hat. Project 1 is identified with the dotted red line and a second project is identified with the blue dotted line.

The matrix report system did not change with the addition of the new platform and Top Hat structure for the functional engineering areas. The only difference between the prior organizational scheme and the new one was the addition of the platform project manager, meaning that some of the functional engineering representatives now have to report dually to both project managers. To avoid a double dotted line report to both of the project managers, all the project representatives (the managers) had their secondary report to the project through the product family director instead.

Vehicle integration represents an additional area for product development. They have the responsibility of integrating all the engineering attributes into the vehicle and making sure the vehicle achieves its planned functional attributes such as the vehicle weight, NVH, overall safety, fuel economy, etc. This area is concerned with the vehicle level attributes instead of the specific subsystem attributes. Because of the difficulty of identifying if a vehicle attribute is Top Hat or platform driven, the vehicle integration manager also had a dual dotted report to both project managers, but organizationally this was achieved with a dotted line to the product family director. To simplify the diagram, the design studio, which is in charge of developing the styling theme and leading the overall product appearance, was also included as a part of vehicle engineering since their role is also related to the vehicle as a whole instead of only a portion of it.

The manufacturing and the marketing functions also have representatives in the project, although they are not part of the product development department itself. The manufacturing area is further divided into two main areas: manufacturing and purchasing. We refer to manufacturing when it is related to the in-house manufactured components and the vehicle assembly; on the other hand, purchasing is in charge of finding and selecting a suitable supply base for the product components. Finally, the project's organization is completed with the finance department, which works closely with product development, purchasing, manufacturing and marketing to ensure the vehicle and the project are compatible with the company's business plan.

The key organizational change was the introduction of the platform and the Top Hat project managers with their respective new roles and responsibilities for integrating their platform and Top Hat projects. Several project managers were assigned to the 12 projects based on the similarity of the product globally. For example, the product "Shepherd" was assigned to project manager B. However, the "Hound" and the "Terrier" products were also assigned to the same project manager because they shared many features with relatively minor modifications required to the original product, the "Shepherd". In this context, if the similarity between products was high, to maintain the synergies between products, a single project manager was assigned. If the exterior differentiation of the products was significant (including different brands), then an additional Top Hat project manager was appointed; nevertheless, the underbody or platform project manager was kept the same to facilitate coordination across waves. A total of 7 project managers (including the platform project manager) were appointed for the overall product family development. These were assigned according to the product family summary in table 3.1.

While the projects required a new Top Hat and platform organization, the engineering functional areas and vehicle engineering operated in a similar structure under the new organization. Yet, their reporting system now became more complicated since the project representatives have to report dually to two project managers, instead of one as done in the past. The mapping of the new structure was not completely disruptive since many of the project representatives still had a one-to-one project manager report. For example, chassis or powertrain reported to the platform project manager, and the body interior department reported to the to the Top Hat project manager. In a similar way, other integration related areas such as marketing, finance and purchasing also had a dual report to the Top Hat and platform projects. In most cases, they were able to differentiate themselves in the same two areas to support the new organization.

In summary, the key difference in the organizational chart with other product development organizations (for a detailed discussion of product development organizations in the world auto industry see Cusumano and Nobeoka, 1998) is that the projects are now identified by two rows: the platform and the Top Hat rows. The new organization was not disruptive to the development areas, but it was disruptive to the integration areas which in many cases had to either adapt and assign a representative for the platform project and one for the Top Hat project or accept the dual report system with the additional burden to their activities as they have now to report to two different project managers.

3.4. Projects Governance

3.4.1 Product Development System

Every vehicle development project in the company was managed using a common product development system that details all the processes, tasks, milestones, gateways, prototypes, tools and guidelines utilized by the organization to deliver new vehicle products into the market. The product development system is the result of several years of learning from prior projects with the overall aim of reducing time to market, lowering product development costs, and improving product quality.

The product development system covers all the development phases from pre-project planning to mass production, and follows a similar logic of the generic product development process (PDP) as proposed by Ulrich and Eppinger (2008). Each major PDP phase is followed by a major project and product review, representing a decision point (milestone) to continue to the next development phase. The Major product and project reviews occur at the beginning of each of the phases detailed in figure 3.1. Besides covering all the different development phases, the scope of the PDP includes also the different organizations that interact in product development: product engineering, marketing, manufacturing, purchasing, finance and project management, and focuses on the cross functional integration and collaboration of all these areas to achieve the product and project goals.

All of the functional areas represented in the project were not only key contributors to the project, but also main project stakeholders, who along with the Board of Directors and the governmental agencies defined all the product requirements. In each of the major project reviews, high level project compatibility with original targets was reviewed and it was decided if the project should continue to the next phase. These reviews represented the main governance method to ensure the project was being developed as planned, and during these reviews, major plan updates were approved, including different product functional attributes, different financial targets or any project delay to the original plan. In this particular case study, these project reviews were bundled by platform wave: all the different derivatives for each wave were assessed concurrently to ensure a balanced execution of all projects.

Two specific characteristics of the product development system are relevant to the case study. First, all the development phases and overall project development time are defined according to standardized guidelines based on prior project's experience. The target development time is based on the relative change magnitude compared to the replaced or a comparable vehicle. Changes (either major, minor or carryover design) are assessed for the exterior and interior, the chassis and suspension, and the engine and transmission designs. The second attribute is that the product development system positively acknowledges different upperbody and underbody differentiated workstreams as well as integrated workstreams for the overall vehicle. These two features allow for better acknowledgement for product families and product platform developments as they can not only accommodate the specific underbody development time based on the relative change level, but also execute it independently while the vehicle exterior and interior styling is still undefined.

The standardized product development system and the differentiation of the upperbody and underbody development workstreams are clear indications that the company has effectively incorporated the product platform and product family approach into their product development process and therefore has enabled senior management support. The entire organization recognizes that enabling global platform usage and achieving product commonality are enablers for significant product development benefits.

3.4.2. Product Development Processes / Phases

The product development process is detailed in figure 3.6, which reflects the same phases as in figure 3.1. The figure shows when the development process is divided into the underbody (figure upper portion) and upperbody (figure bottom portion) design streams for later integration. In the middle blue portion, common vehicle project reviews are included where the overall vehicle business case and compatibility are verified. The vehicle projects start after the product planning phase is finalized and ends when mass production initiates.

The product planning phase can be considered as a part of the ongoing business cycle of the company. In this phase, an opportunity is identified and the business and product proposition are studied in depth for its engineering feasibility and profitability. If it is proven successful, then the project is authorized for its development and the formal project starts. During the product planning portion the following activities are performed:

- Vehicle architecture and major product performance targets are established
- Specific functional design and sourcing strategies are rolled into the overall plan (commodity plan)
- Manufacturing sites are proposed
- Financial targets for development, variable costs, and production investment costs are settled
- Project schedule is finalized, including the date for starting production.

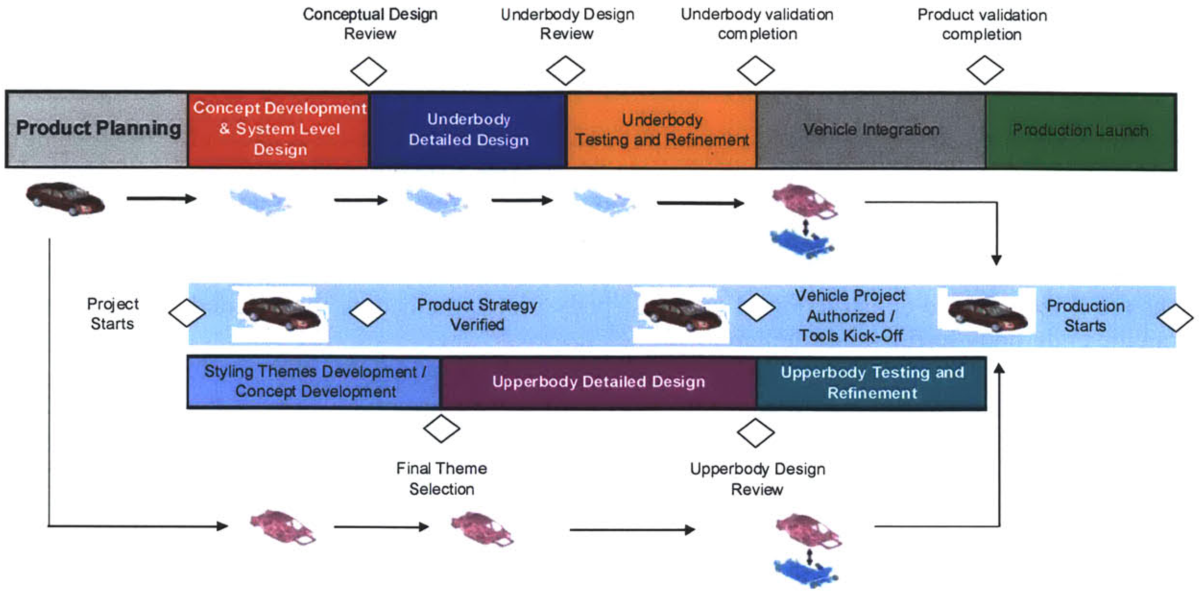


Figure 3.6: Product development system processes

The product planning phase is usually performed by specialized departments and by a relatively small team to allow several business proposition iterations to derive the final proposal and the project charter detailing all the different targets and assumptions. This document represents the starting point for the project after the overall strategy is authorized by senior management.

After the product planning phase ends, the formal project starts and its development is appointed to the development team composed of representatives from all the different affected departments. In the first phase, conceptual designs are developed by the project team, which are later reviewed and selected based on the needs of the project and the product. The overall project, product, and business proposition compatibility are re-evaluated compared to the initial proposed targets as proposed in the project charter. The focus of the product development system is to find a balanced proposition where the product can meet both the business and functional targets, and if not, then the design should be revised until a balanced combination is found. The process should be driven by compatibility verified by all functions instead of a quick design completion.

The main deliverable from the conceptual design phase and the project strategy verification review is ensuring that the overall vehicle targets (cost, investment, functional attributes, weight, etc), known as top down targets, can be met by breaking them out into their respective system or component targets, known as bottom up targets. To achieve bottom up to top down compatibility, the development team propose their respective system and component designs and estimate their functional attributes and costs. After all the designs are finalized, the integration activities take this information and compile the overall product perspective by adding up the contribution of all the different vehicle components. If any incompatibility is found, the integration team requests the development team to modify their proposals and the design iterations continue until compatibility is achieved and the final concept is selected.

After the first round of design concepts are proposed and revised for their overall product compatibility, they are selected to be the main design alternative. These designs are used as the first design proposal to find a suitable manufacturing source. In parallel to the compatibility iterations, the designs are simultaneously being quoted to prospective suppliers to further validate the engineering costs and functional targets. Also, during this phase the vehicle BoM is populated by the development engineers that must translate the original project charter intent into detailed component part numbers that correlate to the latest design proposal. The bill of materials becomes one of the major project status repositories and is shared with all the different affected parties. Once the project has achieved compatibility, then the process is broken down into underbody and upperbody development.

During the underbody detailed design phase, the development engineers focus their efforts on finding a suitable manufacturing source for their components and also on progressing the engineering drawings and underbody mockup. During this phase, the proposed designs are continuously being verified with virtual engineering tools like CAD, CAE and CAM and are constantly updated to achieve a robust design proposal for each component. After the underbody achieves complete geometrical compatibility, the designs are frozen for the construction of the underbody prototype, ending the detailed design phase and starting the underbody testing and refinement phase.

In the underbody testing and refinement phase, the underbody prototype is built and later tested and validated for its proposed functional targets. As the prototype is being verified, all types of engineering issues are found and the designs are updated to achieve all the functional requirements. This phase is finalized when all the underbody testing is completed and all the corrective plans are in place to achieve the project's targets. The finalization of the underbody testing completes the independent underbody workstream, and moves forward to the integration with the upperbody of the vehicle.

Parallel to the underbody development, the upperbody also progresses their development. Figure 3.5 details how the upperbody and underbody streams overlap in the development. The upperbody workstream differs significantly because in addition to the desired functional attributes, the appearance and style of the product become the major sources of the design churn. To achieve compatibility the functional attributes and the cost of the parts have to be balanced with their appearance. The design studio leads all the exterior and interior appearance proposals, which are later evaluated and traded off by the engineers that must deliver the cost and the functionality of their parts. Conceptually, the vehicle styling decision should be delayed as much as possible to achieve a competitive product, so multiple design proposals or themes are evaluated and developed until a single theme is defined.

Once a single theme is defined, the upperbody detailed design phase starts. Here, the upperbody engineers perform similar tasks as the underbody engineers: progressing their engineering drawings, evaluating their functionality with virtual tools, achieving geometrical compatibility with the upperbody and the underbody, and also selecting their manufacturing sources to ensure that the proposed designs are feasible from a manufacturing perspective. This phase finalizes when geometrical compatibility is achieved. Once the design is completed, it is sent to the selected supplier for a final agreement. This same design is frozen for the construction of the complete vehicle prototypes.

At this point, the underbody should be validated and the upperbody design should be completed and compatible. This represents one of the major decision points for the compatibility of the project since now all the designs are available and most uncertainty has vanished through the development process. If the project is authorized, then the project will change from a development or planning mode to a delivery mode, where all the targets (now compatible) now become objectives for the product. The following activities will be performed after project authorization:

- The production tools will be kicked off to support future mass production;
- The design will be frozen for the production intent prototypes;
- The underbody workstream will be focused on integration with the upperbody;
- The upperbody will continue their testing and refinement phase, and will also focus on integrating their designs with the underbody.

The last phase of the project represents the production launch phase. In this phase, all the production prototypes should have completed their testing, and all the design flaws are corrected as they are being found for the overall product refinement. In this last phase, the upperbody and the underbody differentiated phases no longer exist and the overall vehicle is the focus of the project. This phase is the last chance the engineers will have to update their designs, and these should be relatively simple design corrections. The focus of this phase is preparing and refining all the production tools to achieve the desired quality for the product. In this final phase, the intensity shifts from design to manufacturing activities.

With a more detailed description of the development phases and processes, it can be concluded that the overall development process is driven to ensure compatibility and avoid useless design rework cycles. Breaking down the underbody and the upperbody development is a key feature which allows product compatibility to converge faster by reducing the scope of the desired compatibility to the upperbody or the underbody in isolation. Finally, while the description of these phases is sequential, each of these phases contains design iterations. As design uncertainty is reduced by evolving the vehicle design, these design iterations may or may not have impact on the cost or the functional attributes. To manage these design iterations, a change control process was also managed within these stages and will be discussed in the next section.

3.4.3 Platform Change Control

It is clear that the major product reviews detailed by the product development system are not sufficient to ensure that the development is converging into a balanced tradeoff of functional attributes and costs. While the day-to-day project progress was tracked in a weekly progress meeting, a change control meeting process was also required to discuss project and product tradeoffs and make relevant design decisions in a timely manner. The change control meeting had a cross functional scope and it was the major trade-off forum where the project manager defined the solution for issues that were brought to this forum. Examples of topics that were brought to this forum are as follows:

- Engineering changes that required a relatively significant variable cost increase or investment. This represented the majority of the topics of the change control meeting
- Resolution of cross-functional issues where an overall project perspective was required to solve the issue
- Discussion of project timing and support to the product or engineering reviews or prototype builds
- Discussion of product attributes and required engineering changes to achieve their target
- Discussion of cost updates driven by new quotes and driving significant cost increase

The meeting's main focus was to make engineering decisions and discuss engineering changes that required the attention of the platform project manager. However, the scope of the meeting was broadened to discuss other issues similar to the ones above: cost updates, timing updates and other type of reviews. The main output of the meeting was intended to be a decision on what to do based on the available alternatives; however, the discussion was often bounded to a simple status review. The meeting agenda was structured to use the same platform / Top Hat logic as discussed earlier. Depending on the systems affected by the decisions, each discussion was directed to the appropriate project manager or sometimes scheduled for a joint discussion if the scope included both major subsystems. The change control forum represented the ultimate step of escalation inside the project. If alignment or direction was not achieved, then the product family director lead the resolution with the affected engineering director(s). Yet, this represented the minority of cases since most of the discussions were effectively managed in the change control forum. The Top Hat team had also had their own change control forum if they required decisions from the Top Hat project manager in situations where the underbody was not affected.

The platform change control meeting operated in the same manner throughout the scope of the case study. Throughout the development, an independent forum for discussing underbody changes was necessary and eventually the audience required was reduced from two or three project managers at the beginning of the project (concept development phase) to discussions requiring only the platform project manager. This same trend was repeated when the second wave was developing their design concepts. This evolution is shown in figure 3.7 that illustrates how many discussions (most of them engineering changes) with the platform project manager were required. These discussions are divided into those requiring concurrence of the platform project manager in isolation, either for one or more waves; and how many discussions required a joint decision, either with the wave 1 - Dog project managers or the wave 2 – Wolf project managers.

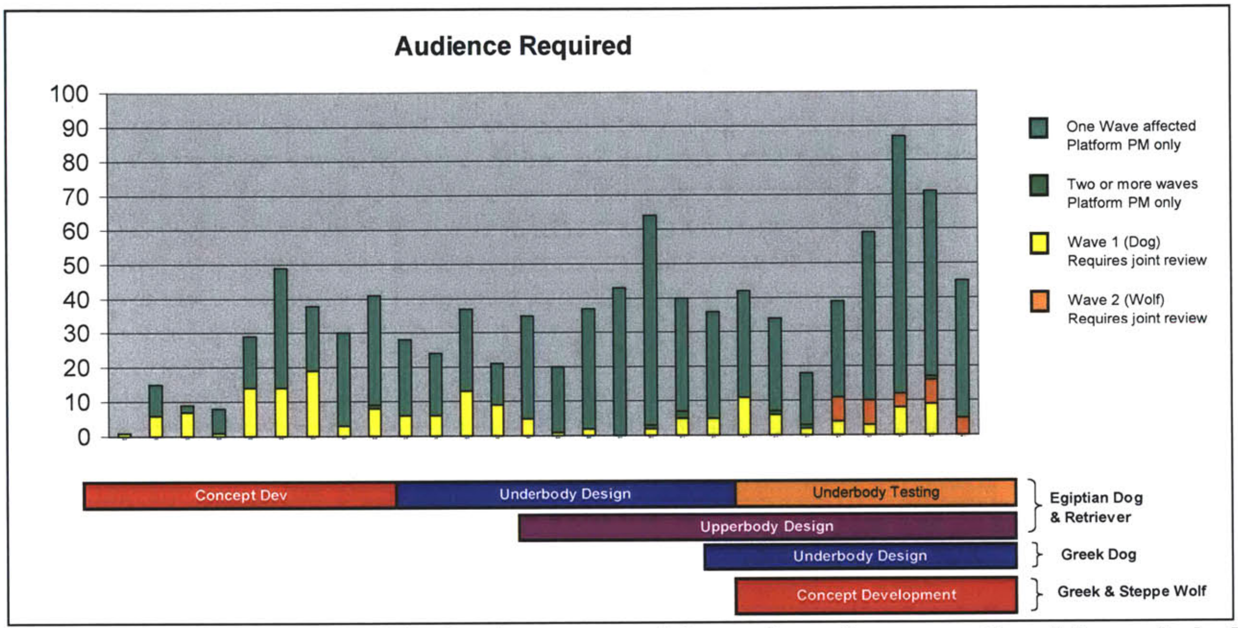


Figure 3.7: Audience required in the change control forum from the perspective of the underbody. As the project progresses, the platform project manager worked in relative isolation with some marginal coordination with the Top Hat project managers.

Another key insight from the figure above is the workload trend. As the project progresses, the number of changes increases. Yet, as a development phase approaches its end, the figure has a peak that reflects the efforts to freeze the design changes and allow other activities to complete their work and integrate the overall vehicle. All these peaks were concurrent to the design freeze dates for any of the projects.

Figure 3.8 reflects all change control discussions from another perspective. The graph shows the scope of change control meeting discussions, and if they were confined to one, two or three waves. The figure shows that the first wave required more attention than the second one, and the number of discussions that were related to more than one wave were even less. The graph illustrates that the synergies of the product platform are diminished if lifecycle offsets exist within the different product family derivatives. While it was intended to have a high degree of commonality and coordination, executing the platform design with lifecycle offsets represents a significant challenge to achieve coordination and commonality. Figure 3.9 is the same graph, but it is divided by the type of discussion if it was either a simple status review, a cost update, or an actual engineering change.

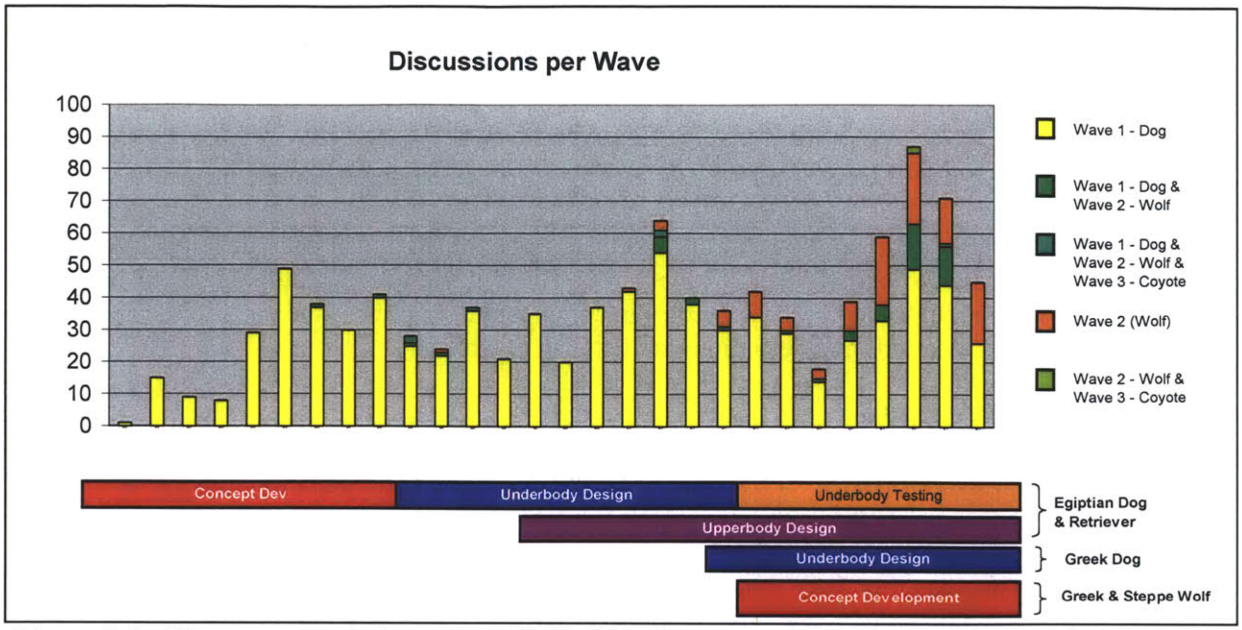


Figure 3.8: Change control discussions divided by the scope of the discussion by derivatives waves.

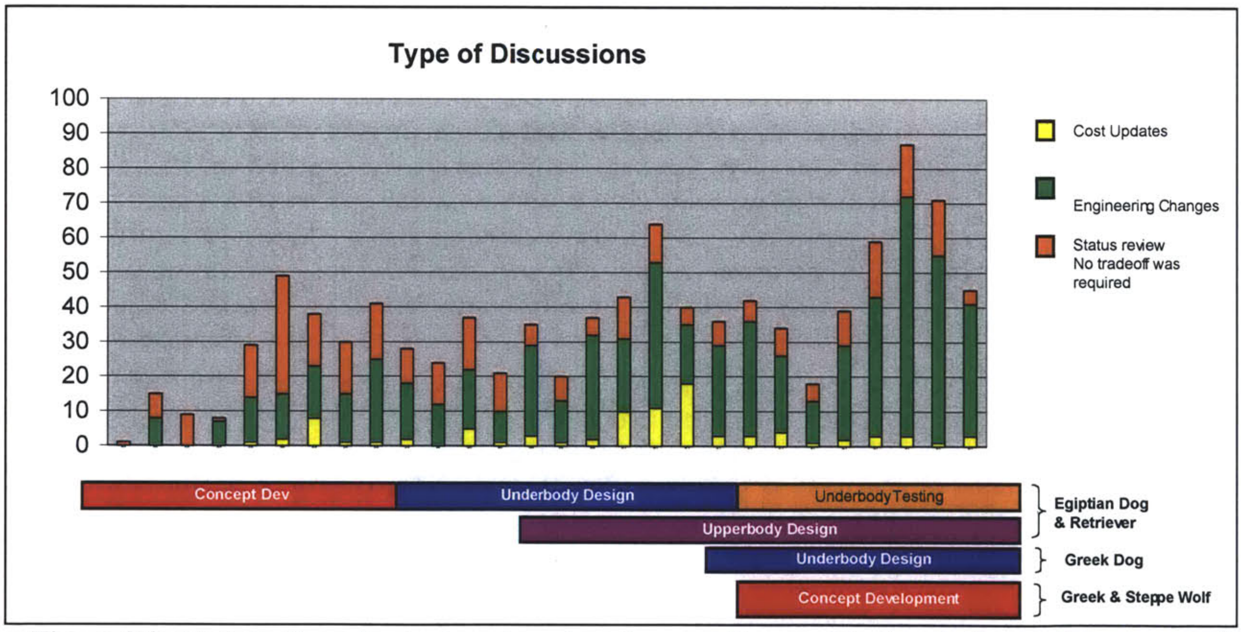


Figure 3.9: Distribution of actual engineering changes discussed in the platform change control.

The change control meeting was one of the most successful features of the platform and Top Hat structure. Since the changes were driven by the functional engineering areas instead of the integration, it was relatively simple to direct them to the appropriate project manager for change approval. Throughout the project, the need of a stand alone platform (underbody) change control meeting was clearly established, although, some coordination was still required with the Top Hat projects. Also, since the meeting was more functionally driven rather than cross-project coordination driven, the original objective for coordinating the three waves simultaneously was marginally achieved since the majority of the changes were assessed for the most urgent wave and the impact for future waves was not discussed as a direct result of the lifecycle offset. The cross-wave coordination increased after the second wave started their conceptual development phase, nevertheless, the low product commonality between products in different waves can be another explanation for the low cross-waves interaction, and this will be explained in the next chapter.

3.5. Product Commonality

3.5.1 Commonality Measurement and Report Method

To measure product commonality, or how similar a vehicle was compared to already developed designs, a simple method was created to classify each of the parts of the vehicle bill of materials. Four basic part classification types were identified to assess the design commonality of each of the parts in the BoM. The basic commonality classifications and definitions are detailed in table 3.4.

Commonality Code	**Definition**	**Criteria**
C	Carryover or Common (Reused)	The part must have the same part number
M	Modified (Tunable)	The part must have a minimum of 70% of tooling or engineering cost reused from another part
N	New Shared Part	The part will have a new part number and future vehicles must be identified that will reuse the exact same part number
U	New Unique Part	The part will have a new part number and will only be used on this particular vehicle

Table 3.4 – Commonality Codes

From the classification above, the "C", "N" and "U" parts have straightforward definitions and only by knowing the part numbers and the vehicles in which they are used, anyone would be able to classify the parts in the bill of materials. However, the "M" (modified or tunable) parts require some engineering judgment to classify and understand if the part is reusing a significant portion of an already developed part modified to new vehicle requirements. Product commonality was reported as the cost weighted percentage for each of the four classifications above.

To better understand the scheme above, the rules for assigning part numbers required the engineers to have a number for any part with slight modification to the design; this was applicable to design revisions or to part variations driven by different features. For example, if one vehicle requires a stamped part to have a hole, and another similar vehicle does not, then the part will have two different designs and therefore two different part numbers. If the shape of the stamping changes, then both parts will be have to change to the latest design. The stamped part example provides a good illustration of what a "modified" part would be, it requires some engineering and tooling development, but it is small compared to the prior stamping shape that has been already developed.

Given that the part classification required knowledge on how many design versions existed, what was the latest design, and how common a variant part was to a similar part, the design engineers were the individuals responsible to classify each of their parts in the BoM. This information was mandatory for each part populated in the corporate BoM system. To classify their parts, the design engineers followed the decision flow diagram in figure 3.10 below. In the context of the case study, a part number represented a complete part or subassembly that was shipped from the manufacturing or fabrication source to the assembly plant. Also, launch timing affected the classification of the parts; if a new part was shared with other derivatives, only the first derivative designated it as new and shared (N), and all the rest of derivatives that reused the same part reported it as carryover (C).

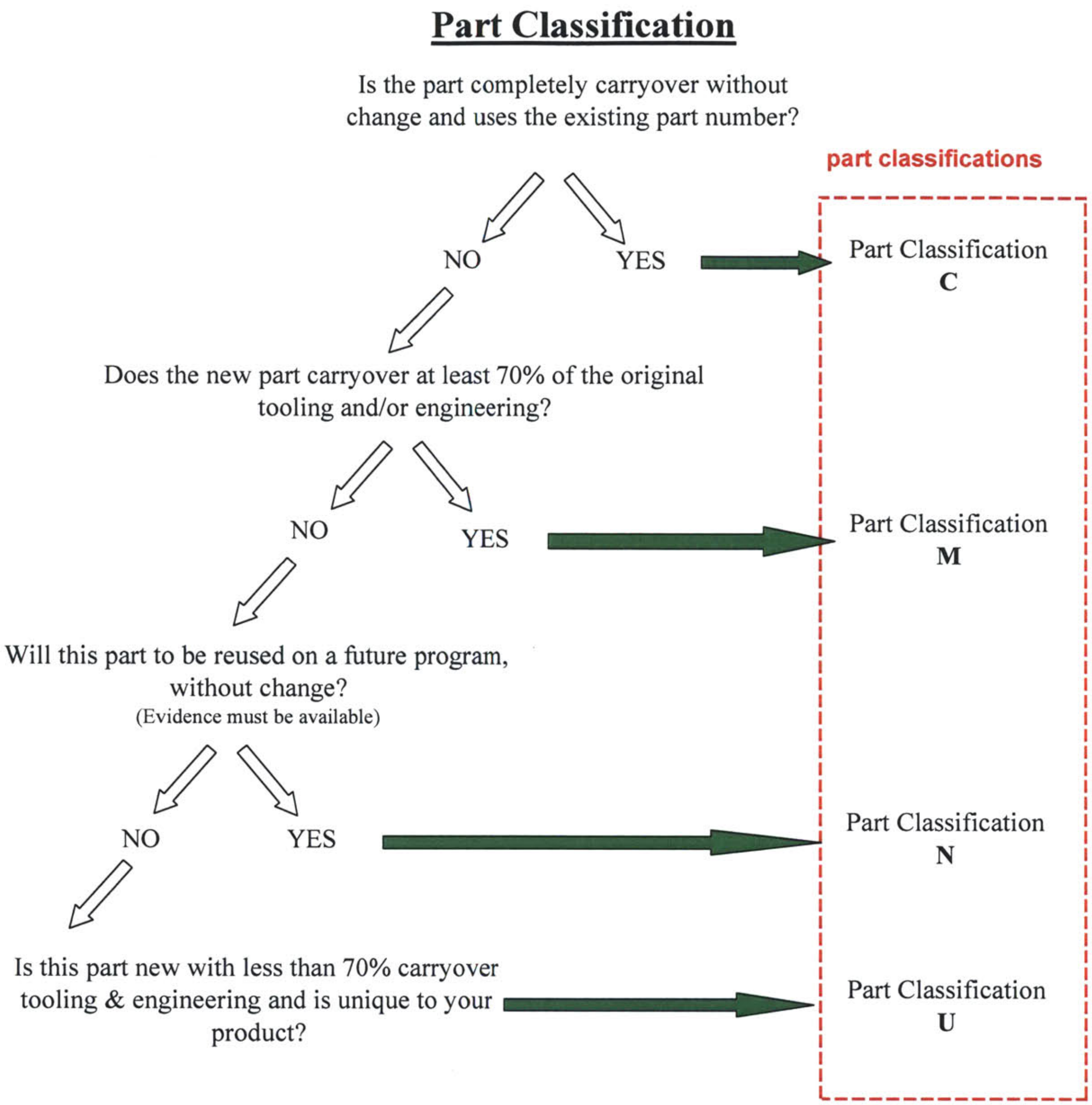


Figure 3.10 Commonality classification decision flowchart.

The decision flow chart above represents how engineering and tooling efficiencies are accomplished in a product platform. If the new product is able to carryover the same design from the prior product, then it will have the maximum benefit ("C" parts). If a complete carryover design cannot be accomplished, then a modification should be pursued to have some efficiency from the prior design ("M" Parts) and avoid major new investment. If any of the above options cannot be accomplished, the next best option is to create a new part that could be shared in various future platform derivatives ("N" parts). Finally, the unique parts may look as the worst scenario compared to the other classifications since they will be used in only one derivative, however, the product platform approach must allow for developing unique parts to achieve product differentiation. Having the right mix of unique and shared parts (how many and what parts) is the cornerstone of the art of platform product development and is context dependent.

In the automotive industry, it is usual to offer a vehicle with multiple features and options so the customer can have a more personalized product. Common options offered in the automotive industry are different engines, transmissions, drive modes (all wheel drive, front wheel drive), body styles (hatch, sedan, wagon, convertible, coupe), radio or navigation systems, interior and exterior colors, leather or cloth trims, wheel sizes/designs, sunroofs and even some styling features (such as moldings, chromed components, badges). In addition, some other options are required for different markets, such as KPH/MPH clusters, right-hand drive and left-hand drive configurations, daytime running lights, dirty fuel capability and so on. This wide range of options and features can create a high number of product combinations that gives customers a wide range of choices, but will also create several part variants that must be designed, developed, tested and managed on the assembly line under normal operations. This array of part numbers is known as part complexity.

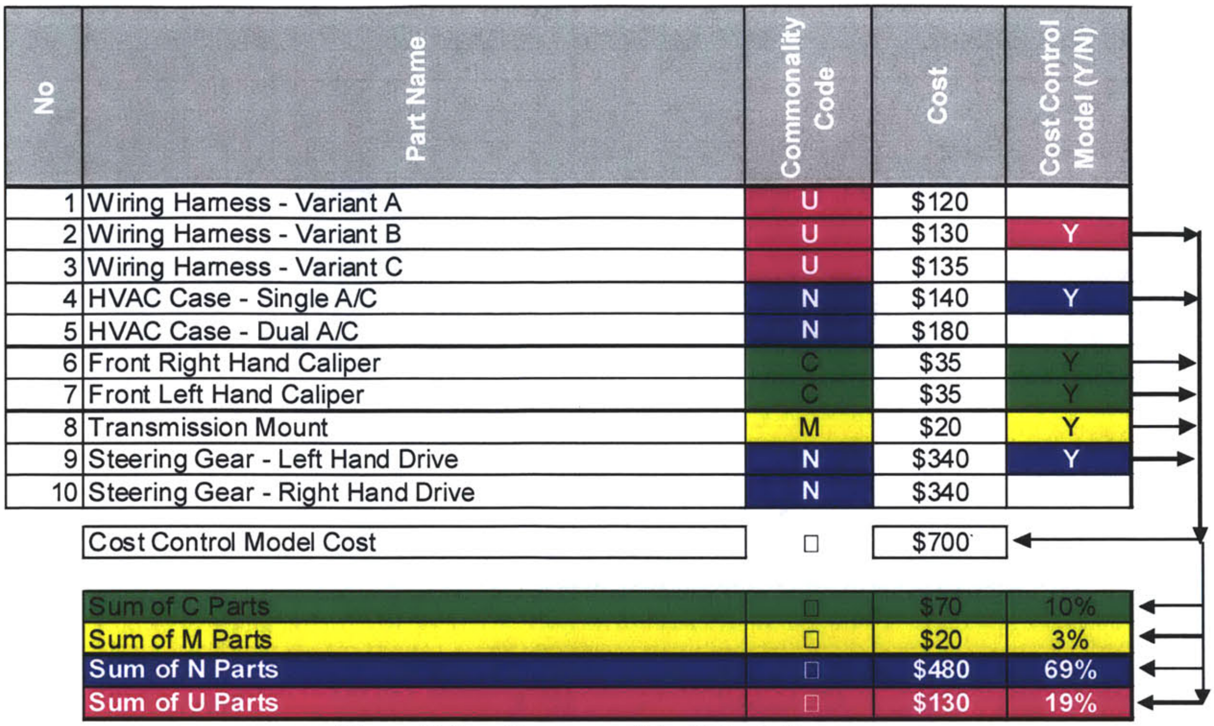
Given the existing part complexity, to understand the cost of the vehicle it was important to refer to a specific set of features and options that build one of the many vehicle combinations and manage the vehicle costs. This combination was known as the "control model" and was selected by the cross functional project team to track the progress of the project. The selection of the control model was usually a common combination (high sales volume) that would be representative of the overall product.

Once all the parts in the BoM were coded by the engineers and the variable cost of each of the parts was available, the project management group used the information to calculate the overall commonality of the product. Product commonality was assessed by adding the cost for each of the commonality classifications and comparing it on a percentage basis to the cost of the complete vehicle. Another option was reporting commonality by counting the number of parts in each category compared to the total number of parts in the vehicle. The definition of "part" in this context is an "end item" to the assembly plant, so it will have the following characteristics:

- Will represent a line item in the product bill of materials
- Will be assembled by either an external supplier or by an internal source
- Is the minimum level of assembly from the perspective of the assembly plant and will be delivered to the assembly line for final production.

The percentage contribution of the variable cost of the parts was selected as the preferred method to report commonality since in many cases the cost of a part can be representative of the effort required to design it. Given the definition of an "end item" above, the span of part costs can fluctuate from thousands of dollars to cents, and it would not be fair to compare a carryover engine assembly (which commonly can represent over 15% of the cost of the vehicle) to a carryover fastener. This could also be considered a disadvantage if a slight modification to a carryover engine was required, since it should not be coded as carryover if the part number changes. Figure 3.11 illustrates the commonality assessment process.

Product commonality was reported in the major product reviews at the end of each of the major design phases. The commonality metrics were compared to their targets based on the degree of change of the product contrasted to its predecessor and to the pre-defined targets from the product development system. To define these targets, first a qualitative level of change was defined and agreed upon by the organization, and then the targets were determined by the product development system based on prior vehicle experiences. Targets were set for carryover and unique parts only but not for shared or modified parts.



No	Part Name	Commonality Code	Cost	Cost Control Model (Y/N)
1	Wiring Harness - Variant A	U	$120	
2	Wiring Harness - Variant B	U	$130	Y
3	Wiring Harness - Variant C	U	$135	
4	HVAC Case - Single A/C	N	$140	Y
5	HVAC Case - Dual A/C	N	$180	
6	Front Right Hand Caliper	C	$35	Y
7	Front Left Hand Caliper	C	$35	Y
8	Transmission Mount	M	$20	Y
9	Steering Gear - Left Hand Drive	N	$340	Y
10	Steering Gear - Right Hand Drive	N	$340	
	Cost Control Model Cost	□	$700	
	Sum of C Parts	□	$70	10%
	Sum of M Parts	□	$20	3%
	Sum of N Parts	□	$480	69%
	Sum of U Parts	□	$130	19%

Figure 3.11 – Commonality Calculation process.

The only caveat on the process above is that the existence of a representative BoM may be a luxury in early product development (planning) phases. In this case, the methodology was basically the same as the one described above; however, the assessment was not done at the individual part level but on a subsystem level. All the information regarding the commonality of the different vehicles that are part of the platform must be clearly stated in the main project charter book, which contains the official project information that translates the customer's needs into high-level engineering requirements. The project charter book includes not only the commonality information, but also a brief description of the changes that would be performed to the product to achieve the product's desired performance. The project charter is a live document that is constantly being updated as the design progresses.

3.5.2 Product Variants Commonality and Divergence

Using the agreed method to report product commonality, all the different derivatives reported their commonality status at every major program review. The following tables (table 3.5a to table 3.5f) illustrate how commonality progressed from the start of the project to the end of the underbody testing or to the end of the upperbody design phases assessing the overall vehicle. Unfortunately, there was no data available for all the vehicles at all the different product development phases.

Shepherd - Egypt				
	Start	Concept Design	Underbody Design	Underbody Validation
C	53.4%	29.0%	26.9%	26.8%
M	11.9%	11.2%	8.8%	4.2%
N	12.6%	50.2%	52.3%	53.6%
U	22.1%	9.6%	12.0%	15.4%

Table 3.5a: Shepherd product for Egypt (First wave)

	Shepherd - Greece			
	Start	Concept Design	Underbody Design	Underbody Validation
C	N/A	66.8%	51.0%	44.4%
M	N/A	1.5%	4.6%	5.1%
N	N/A	27.5%	37.4%	37.9%
U	N/A	4.2%	7.0%	12.7%

Table 3.5b: Shepherd product for Greece (First wave)

	Shepherd - Maya			
	Start	Concept Design	Underbody Design	Underbody Validation
C	N/A	91.9%	82.8%	82.4%
M	N/A	4.9%	2.5%	0.9%
N	N/A	0.0%	9.3%	5.8%
U	N/A	3.2%	5.3%	11.0%

Table 3.5c: Shepherd product for Maya (First wave)

	Retriever			
	Start	Concept Design	Underbody Design	Underbody Validation
C	N/A	41.9%	48.3%	46.3%
M	N/A	15.2%	5.2%	2.2%
N	N/A	7.7%	7.1%	10.7%
U	N/A	35.2%	39.4%	40.8%

Table 3.5d: Retriever (First wave)

	Wolf - Greece		Wolf - Egypt	
	Start	Concept Design	Start	Concept Design
C	53.9%	41.2%	84.4%	56.7%
M	10.6%	16.1%	5.1%	8.0%
N	35.5%	37.1%	5.7%	16.2%
U	0.0%	5.7%	4.8%	19.2%

Table 3.5e: Wolf product for Egypt and Greece (Second Wave)

	Wolf - Maya		Steppe Wolf	
	Start	Concept Design	Start	Concept Design
C	98.8%	78.0%	79.5%	76.3%
M	0.0%	2.7%	9.9%	1.7%
N	0.0%	7.1%	0.0%	0.1%
U	1.2%	12.1%	10.6%	21.9%

Table 3.5f: Wolf product for Maya and Steppe Wolf product (Second Wave)

	Tundra Wolf	
	Start	Concept Design
C	68.5%	54.2%
M	8.6%	8.9%
N	1.7%	1.1%
U	21.2%	35.7%

Table 3.5g: Tundra Wolf product (Second Wave)

Tables 3.5a to 3.5g – Commonality progress by design phase. The first wave of projects shows the progress throughout 3 phases. The second wave of projects shows the change from the initial commonality intent to the commonality status at the end of the conceptual design phase.

The tables above need further clarification to understand the commonality evolution as the projects progress. For every project at their start, the assessment method was performed on a surrogate BoM and assessed by the planning activity. Unfortunately, for the first wave the criterion was different from the Shepherd which included all the different regions, so data is only available for one product. After the project lead changed from the planning organization to the project team, the assessment was done using the same criteria and process, so the evolution reflects the actual changes. Figures 3.12 and 3.13 present a graphical evolution of the data above.

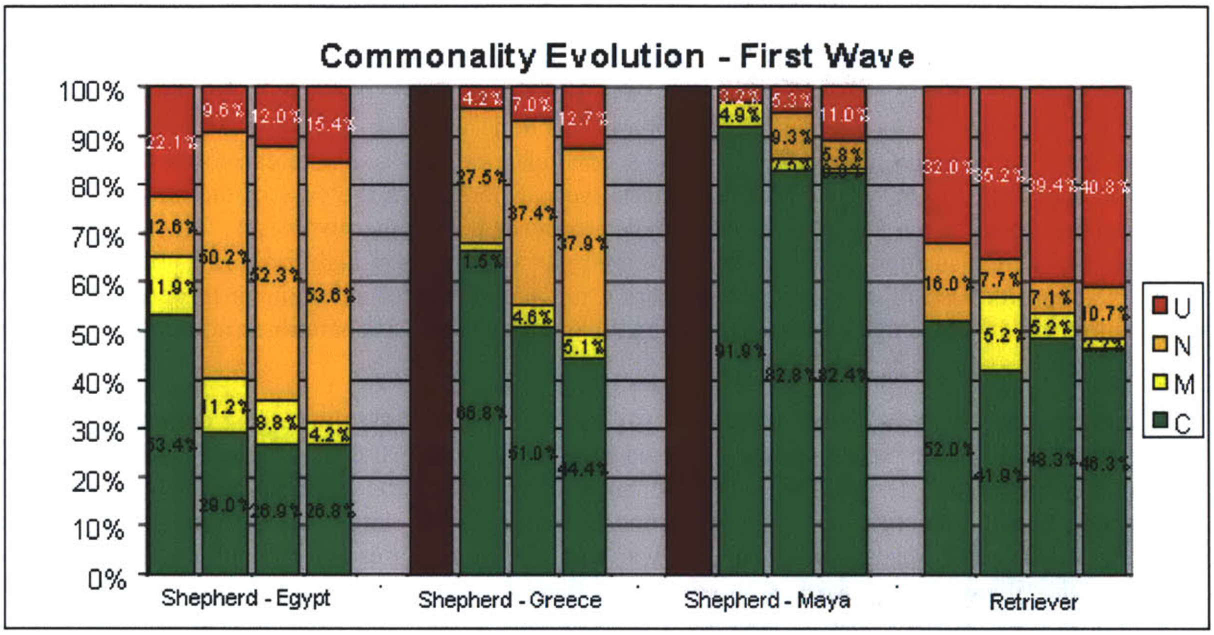


Figure 3.12: First wave commonality evolution for the overall product

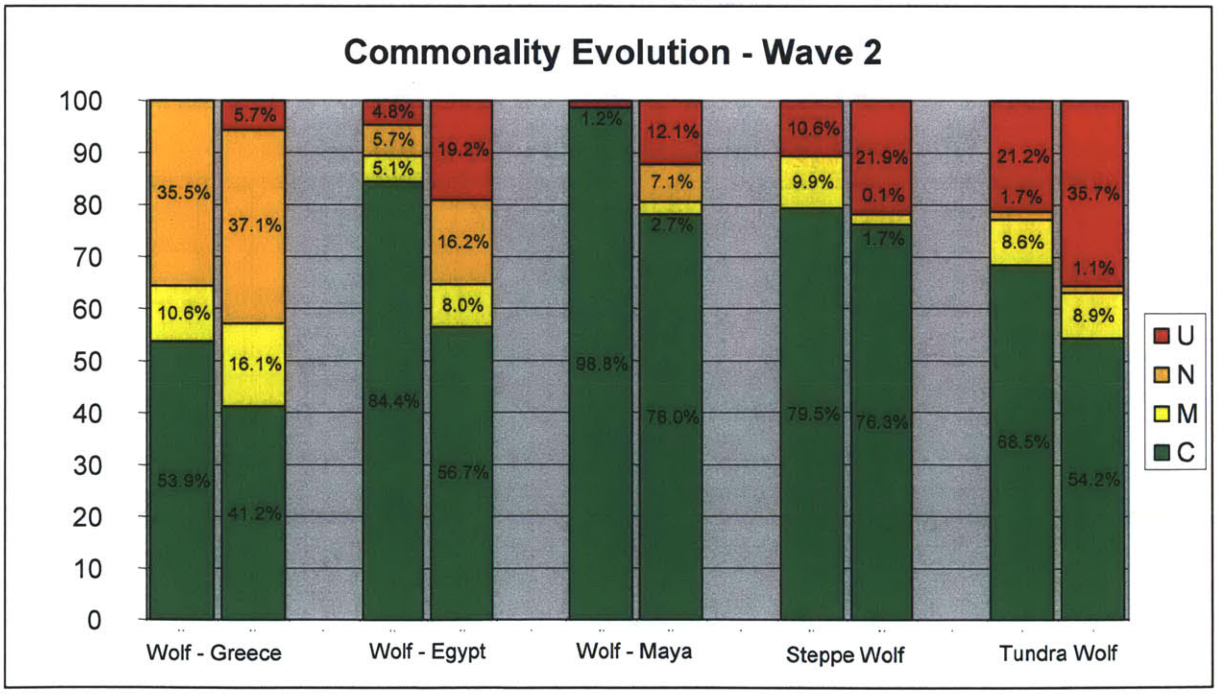


Figure 3.13: Second wave commonality evolution for the overall product

Another disclosure about the information above is that timing was a key feature of the commonality assessment as it was based on the projected production start date of the products. In this case, the first product (Shepherd in Egypt) compared its design to the donor platform or other products already available in the company. The rest of the products were then compared to all of the prior products plus the first derivative, and so on. In the case of the second wave, most of the common parts were based on the first wave derivatives and not on a prior donor platform. As more products were added to the platform, their commonality was expected to increase since there were more vehicles and more parts available to be shared with the new derivatives.

The data presented in the tables and figures above show two clear trends in line with the insights on divergence presented by Boas (2008). First, the amount of carryover or common content was reduced with the progress of the project in all but one case (Retriever); and, the reason for this is that the assessment at the end of the conceptual design phase was done with a surrogate BoM but with greater fidelity than the original planning assumption. The second clear trend was the increase of unique parts for all but the Shepherd in Egypt, whose original criteria were different. In the case of the new and shared parts (N) and the modified parts (M), there was no clear trend as in the other two cases. The reason is that the criterion for being classified as a modified or shared part was not as straightforward as the other two; defining a part as modified requires some engineering judgment, and for an engineer to define a part as shared, requires clear knowledge of the reuse plans for the part, which is sometimes not available to the engineers.

Another important observation about figure 3.12 for the first wave is that the largest commonality loss (carryover parts) and unique parts increase occurred between the original commonality assumptions and the end of the conceptual design phase. As the project progressed the changes in the commonality assessment decreased, reflecting how the attention shifted first from larger parts (higher cost) at the beginning of the project towards smaller parts (lower cost) in future phases and this reflects a fine tuning of the commonality assessment for the smaller parts that could have been optimistically assessed as common in prior reviews.

Following the Top Hat and platform projects convention, the above data was also broken down into their respective Top Hat and platform projects. The following figures (3.14 and 3.15) show the commonality of the platform or underbody, which was expected to be high compared to the Top Hat systems that were expected to differentiate the product and to be unique and different throughout the derivatives.

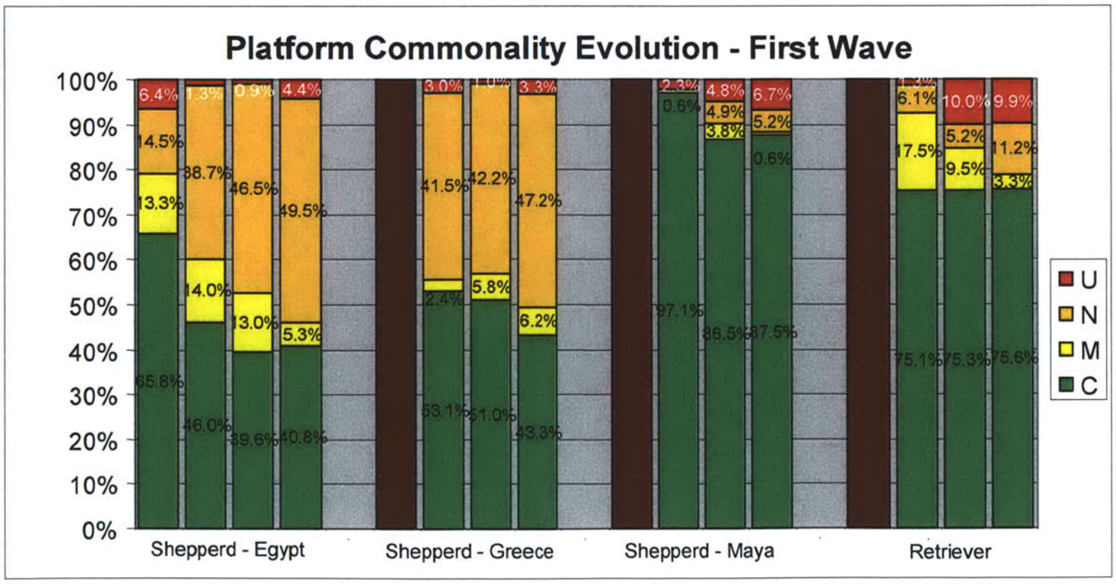


Figure 3.14: First wave commonality evolution for the platform subsystems

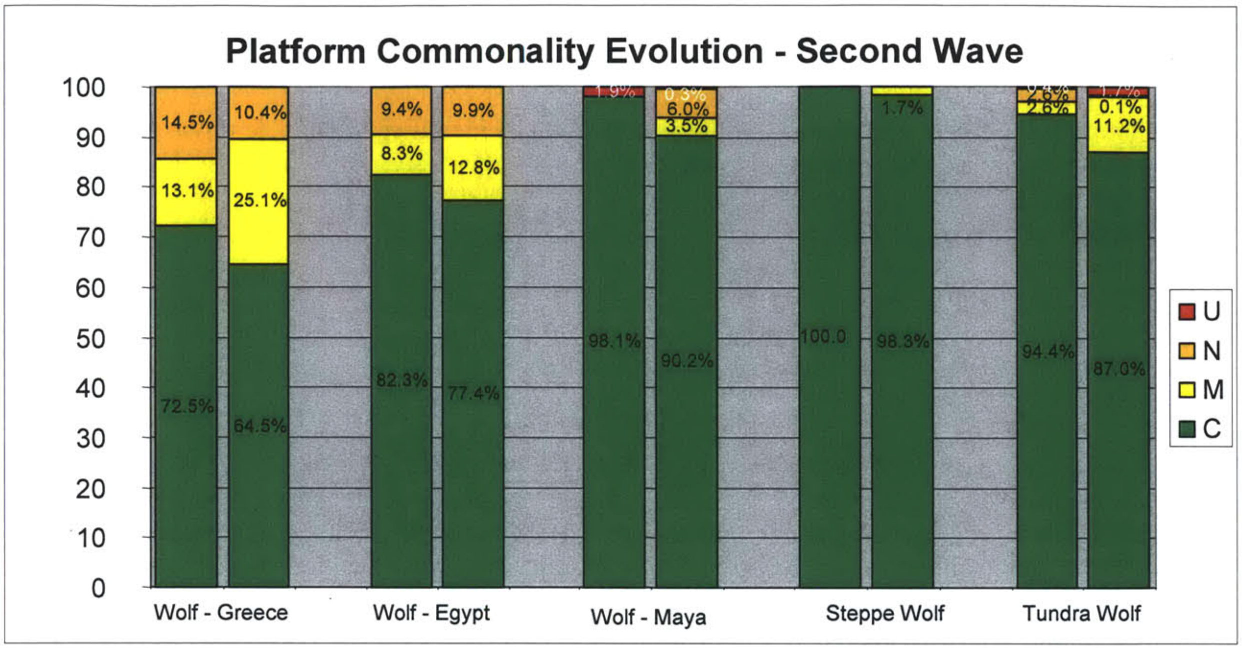


Figure 3.15: First wave commonality evolution for the platform subsystems

The commonality assessment for the platform projects showed a similar trend: an increase of unique parts, a decrease of carryover or common parts, and deeper steps from the beginning of the project towards the completion of the conceptual design phase. However, in some products like the Retriever or the shepherd in Egypt, the assessment was relatively unchanged from the prior assessment. This relative improvement in commonality assessment compared to the perspective of the overall vehicle demonstrated tighter control on commonality that was driven by the new platform organization and the platform project manager. This tighter control was implemented on expensive parts and was effectively reflected as lower divergence compared to the first wave. A key improvement compared to the first wave was the lower divergence for the lead product compared to the planning assessment.

Finally, another two key pieces of the puzzle to understand the commonality progress throughout the project will be addressed in upcoming sections: the individual design changes that drove the divergence behavior above and the relative scope of the commonality study. The studies above reflect the commonality for one build combination and do not include the fact that several engines and transmissions, different equipment levels, and divergent regional requirements also drive unique designs that were not included in the information above.

3.5.3. Part Sharing Studies

To complement the cost weighted commonality process that the company uses, a different approach was used to reflect the commonality of the products in terms of their total scope including all the different part variants that had to be designed to support the market needs. Based on the part numbers available on the official BoM, a part count commonality assessment was performed to further understand how the project was sharing the designs across the different derivatives.

The methodology to perform the sharing studies was simple: using the existing bill of materials, all the available part numbers were cross checked to identify what vehicles were using the exact same part number in their BoM, regardless of the quantity per vehicle. The subject parts represent the manufacturing perspective, and each part represents a complete assembly or "end item" that is shipped to the assembly plant and assembled in the vehicle. The objective of the study is to understand how the individual parts were being shared, excluding the fasteners from the study (which are low value with high expected reuse across the vehicles). One part can represent the complete engine, a steering wheel or a small switch, and the percentages represent the fraction from the total parts count. There were 7 existing BoMs in total for the first 2 waves, each of these BoMs will be identified with an acronym (refer to table 3.1 for additional details on the projects below):

1) Egyptian Dog Project (ED) – Wave 1
2) Retriever Project (R) – Wave 1
3) Greek Dog Project (GD) – Wave 1
4) Mayan Dog (MD) – Wave 1
5) Greek Wolf & Steppe Wolf (GW) – Wave 2
6) Egyptian Wolf & Tundra Wolf (EW) – Wave 2
7) Mayan Wolf (MW) – Wave 2

As seen in the list above, the strategy for assigning BoMs to products / projects was not consistent within waves. The first wave had a BoM for each project, while the second wave had one for each manufacturing plant / region. The strategy for assigning BoMs to projects should not affect commonality output, however, maintaining fewer BoMs could increase product commonality since it would represent less work for the design engineers to maintain common parts for all the derivatives (uploading a change 3 times 5 times) and for easier understanding at the manufacturing plant. Having a common BoM could easily provide insights since the information would not need to be validated across several BoMs as performed in the subsequent studies.

Because the parts sharing studies were not an element of the standard project status report, these studies were not performed at the end of a design phase, but were performed randomly throughout the project. Figure 3.16 details the approximate time when these studies were preformed. The results of these sharing studies are also shown in tables 3.6 and 3.7 representing the sharing analysis for both waves. The tables are divided into all the possible sharing combinations among the different derivatives for wave 1 and wave 2. In total, three sharing analyses were performed for the first wave and two sharing analysis were performed for the second wave.

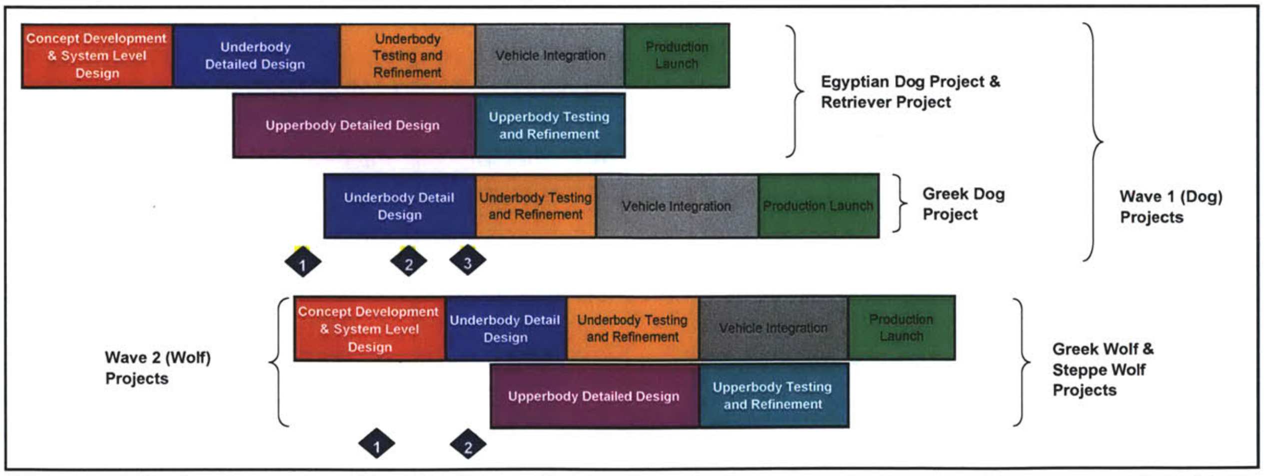


Figure 3.16: Approximate timeline when the sharing studies were performed for wave 1 (3 studies) and wave 2 (2 studies). See blue diamond symbols.

				Assessment 1		Assessment 2		Assessment 3	
Unique Parts	Egyptian Dog (ED)			10.8%	60.2%	10.7%	68.2%	10.1%	71.4%
	Greek Dog (GD)			30.0%		37.6%		40.2%	
	Mayan Dog (MD)			4.3%		3.9%		4.5%	
	Retriever (R)			15.2%		15.9%		16.7%	
Used in 2 Derivatives	ED	GD		2.9%	16.5%	2.1%	11.5%	2.0%	11.3%
	ED	MD		1.0%		1.4%		1.3%	
	ED	R		2.9%		3.4%		3.0%	
	GD	MD		9.3%		4.0%		4.5%	
	GD	R		0.3%		0.6%		0.5%	
	MD	R		0.0%		0.1%		0.0%	
Used in 3 derivatives	ED	GD	MD	10.3%	14.5%	8.7%	13.6%	8.5%	11.7%
	ED	GD	R	3.1%		3.7%		2.3%	
	ED	MD	R	0.6%		0.9%		0.5%	
	GD	MD	R	0.6%		0.4%		0.4%	
Shared for Wave 1	All wave 1 - Dog Products			8.8%	8.8%	6.7%	6.7%	5.5%	5.5%
Totals				100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study				Baseline		21.0%		0.5%	

Table 3.6: Sharing analysis for complete vehicle, first wave of derivatives.

			Assessment 1		Assessment 2	
Unique Parts	Greek Wolf & Steppe Wolf (GW) - Greece		15.1%	55.4%	32.6%	68.2%
	Egyptian Wolf & Tundra Wolf (EW) - Egypt		32.1%		27.4%	
	Mayan Wolf (MW) - Maya		8.2%		8.3%	
Used in 2 Regions	Greece	Egypt	1.4%	26.4%	1.4%	19.0%
	Greece	Maya	22.7%		13.6%	
	Egypt	Maya	2.3%		3.9%	
Shared for all Wave 2	All wave 2 - Wolf Products		18.2%	18.2%	12.8%	12.8%
Totals			100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study			Baseline		32.1%	

Table 3.7: Sharing analysis for complete vehicle, second wave of derivatives.

The tables above represented the fraction of the total amount of part numbers required to assemble all the possible build combinations for each of the vehicles (for wave 1) and for each of the regions (for wave 2). In this other perspective to measure product commonality, the effect of divergence was also evident. In both waves and throughout each of the assessments, the number of shared parts decreased in each of their different combinations: shared across or shared within two or three vehicles. On the other hand, these parts have therefore increased the number of unique parts. In order to compare the results from wave to wave, table 3.8 for wave 1 is comparable to table 3.7 as it shows the assessment by region.

			Assessment 1		Assessment 2		Assessment 1	
Unique Parts	Egyptian Dog and Retriever (ED) - Egypt		28.9%	63.1%	30.0%	71.5%	29.8%	74.4%
	Greek Dog (GD) - Greece		30.0%		37.6%		40.2%	
	Mayan Dog (MD) - Mesopotamia		4.3%		3.9%		4.5%	
Used in 2 Regions	Egypt	Greece	6.2%	17.2%	6.4%	12.7%	4.9%	11.2%
	Egypt	Maya	1.6%		2.3%		1.8%	
	Greece	Maya	9.3%		4.0%		4.5%	
Shared for all Wave 1	All wave 1 - Dog Products		19.7%	19.7%	15.7%	15.7%	14.4%	14.4%
Totals			100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study			Baseline		21.0%		0.5%	

Table 3.8: Sharing analysis for complete vehicle, first wave of derivatives, by region.

Comparing both tables above, the divergence effect by region was also evident; however, the first wave showed not only more parts shared for all the derivatives, but also a higher number of unique parts. In both tables, there was also an additional effect based on BoM completeness. Also, in both cases the total number of parts increased significantly from the first to the second assessment. Parts were originally missed in the BoM and then corrected and included accordingly. The trend of later adding additional parts increases the number of unique parts, as shared parts are less likely to be missed. The first wave had a larger number of common parts probably driven by the common styling, however, the first wave had more build combinations than the second wave (more engines available), driving a higher number of unique parts.

Another use of these sharing studies was to understand how the parts from first wave were being reused in the second wave. Table 3.9 represents the fraction of parts (based on the total available parts for both waves, excluding fasteners) that were being shared between the affected vehicles. This adds the effect of the first wave (table 3.6) in the rows and the effect of the second wave (table 3.7) in the columns, creating a matrix that reflects the intersection between both waves and representing all the possible sharing combinations for the product family.

Table 3.9 is a complete perspective on how the parts are shared among all the different regions for both waves. The percentages reflect the unweighted fraction of the total parts count for all the products for both waves. The numbers in the table reflect the geometrical complexity of adding more vehicles into the same platform, allowing more combinations of parts. Table 3.9 summarizes at the bottom the overall results comparing the first wave to the second. The results from figure 3.17 can be obtained from table 3.9; however it is easier to understand the results by comparing two groups at a time.

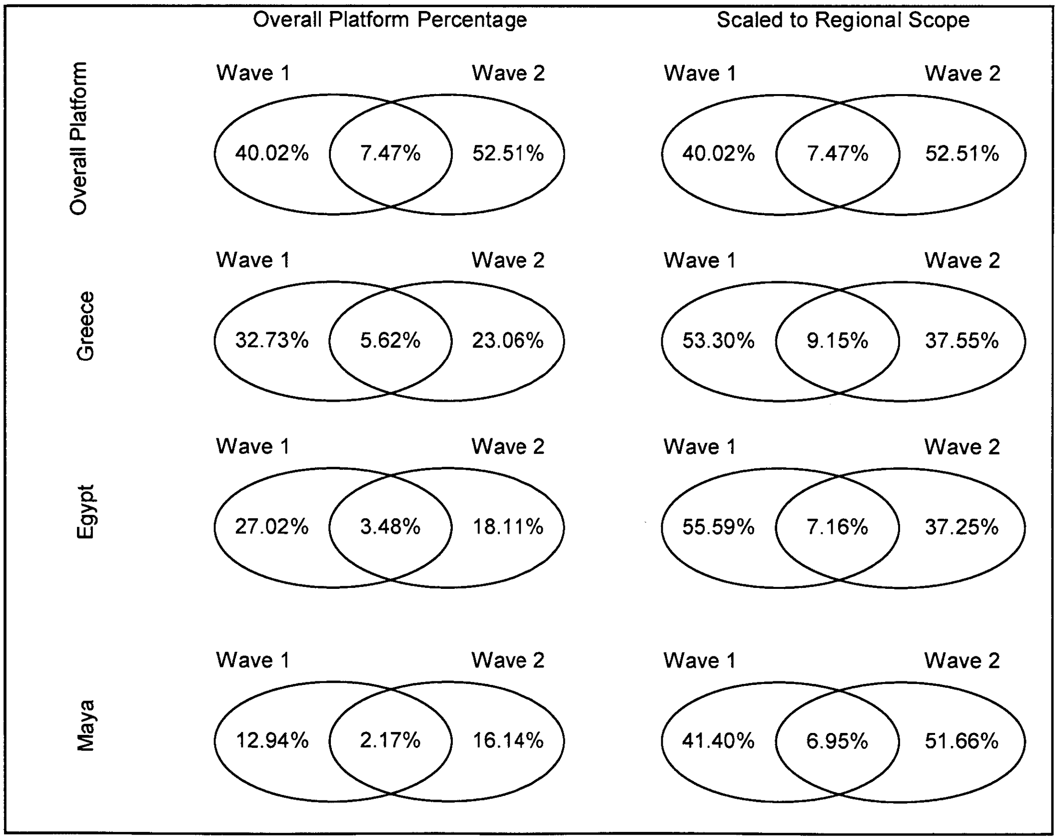


Figure 3.17: Wave 1 to Wave 2 Regional Sharing from a complete vehicle perspective.

				Unique Parts			Used in two Regions			Shared for all Wave 2	Wave 1		
				Greece	Egypt	Maya	Greece Egypt	Greece Maya	Egypt Maya	All Regions	Unique Wave 1	Shared w/ Wave 2	Total
			WAVE UNIQUE	**12.74%**	**12.27%**	**3.66%**	**0.33%**	**5.68%**	**1.68%**	**3.65%**			
Unique Parts	Egypt		**16.58%**	0.15%	0.59%	0.00%	0.02%	0.00%	0.07%	0.45%	16.58%	1.28%	17.86%
	Greece		**21.36%**	1.92%	0.01%	0.08%	0.02%	0.64%	0.00%	0.06%	21.36%	2.73%	24.09%
	Maya		**2.58%**	0.00%	0.00%	0.08%	0.00%	0.00%	0.02%	0.00%	2.58%	0.10%	2.68%
Used in 2 Regions	Egypt	Greece	**2.48%**	0.13%	0.03%	0.00%	0.11%	0.00%	0.02%	0.15%	2.48%	0.44%	2.92%
	Egypt	Maya	**0.99%**	0.01%	0.02%	0.02%	0.00%	0.00%	0.03%	0.01%	0.99%	0.10%	1.09%
	Greece	Maya	**2.21%**	0.32%	0.00%	0.05%	0.00%	0.07%	0.01%	0.06%	2.21%	0.50%	2.71%
Shared for all Wave 1	All Regions		**6.32%**	0.22%	0.08%	0.05%	0.19%	0.09%	0.03%	1.65%	6.32%	2.31%	8.63%
	Wave 2		Unique Wave 2	12.74%	12.27%	3.66%	0.33%	5.68%	1.68%	3.65%			40.02%
			Shared w/ Wave 1	2.74%	0.74%	0.27%	0.35%	0.80%	0.19%	2.38%		7.47%	
			Total	15.48%	13.01%	3.93%	0.68%	6.48%	1.88%	6.03%	52.51%		100.00%

Table 3.9: Wave 1 to wave 2 complete vehicles overall sharing analysis

The conclusion from the information shown in table 3.9 and figure 3.17 is that by comparing both waves only a fraction of parts were shared between the two waves. While 7.47% of the parts were shared in any wave 1 product with any wave 2 product, it was only 1.65% of the total parts were used in all the plants and vehicles under study. After comparing each region, Greece was the region that was shared more in comparison to the others; however it was a low percentage compared to the total number of parts required to build the vehicles.

The results above represent how the overall products were sharing parts; nevertheless, one should expect a significant number of unique parts in order to achieve product aesthetic differentiation. Following the principles of the Top Hat and platform divisions, the same data presented for the overall vehicle will now be presented for the platform or underbody systems, which are expected to be common within the derivatives. Tables 3.10 to 3.13 and figure 3.18 summarize the part sharing studies for the platform or underbody systems only.

				Assessment 1		Assessment 2		Assessment 3	
Unique Parts	Egyptian Dog (ED)			11.7%	58.5%	14.0%	69.5%	12.5%	74.0%
	Greek Dog (GD)			32.8%		40.8%		47.1%	
	Mayan Dog (MD)			3.9%		3.1%		3.8%	
	Retriever (R)			10.0%		11.6%		10.6%	
Used in 2 Derivatives	ED	GD		4.1%	20.9%	2.4%	10.7%	1.8%	10.1%
	ED	MD		1.2%		0.5%		0.3%	
	ED	R		3.6%		4.1%		3.2%	
	GD	MD		11.2%		2.7%		3.5%	
	GD	R		0.7%		1.0%		1.2%	
	MD	R		0.1%		0.0%		0.0%	
Used in 3 derivatives	ED	GD	MD	3.7%	10.9%	3.1%	10.8%	2.8%	8.5%
	ED	GD	R	5.2%		6.2%		4.8%	
	ED	MD	R	1.1%		1.2%		0.6%	
	GD	MD	R	0.8%		0.3%		0.4%	
Shared for Wave 1	All wave 1 - Dog Products			9.8%	9.8%	9.0%	9.0%	7.5%	7.5%
Totals				100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study				Baseline		38.0%		-8.2%	

Table 3.10: Sharing analysis for the platform or underbody systems, first wave of derivatives.

			Assessment 1		Assessment 2	
Unique Parts	Greek Wolf & Steppe Wolf (GW) - Greece		14.5%	46.7%	37.1%	63.9%
	Egyptian Wolf & Tundra Wolf (EW) - Egypt		23.3%		18.2%	
	Mayan Wolf (MW) - Maya		8.9%		8.6%	
Used in 2 Regions	Greece	Egypt	0.6%	33.7%	2.2%	21.1%
	Greece	Maya	28.0%		10.0%	
	Egypt	Maya	5.1%		8.9%	
Shared for all Wave 2	All wave 2 - Wolf Products		19.6%	19.6%	15.1%	15.1%
Totals			100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study			Baseline		25.4%	

Table 3.11: Sharing analysis for the platform or underbody systems, second wave of derivatives.

			Assessment 1		Assessment 2		Assessment 3	
Unique Parts	Egyptian Dog and Retriever (ED) - Egypt		25.4%	62.1%	29.6%	73.6%	26.3%	77.2%
	Greek Dog (GD) - Greece		32.8%		40.8%		47.1%	
	Mayan Dog (MD) - Maya		3.9%		3.1%		3.8%	
Used in 2 Regions	Egypt	Greece	10.0%	23.7%	9.6%	14.0%	7.8%	12.2%
	Egypt	Maya	2.5%		1.7%		0.9%	
	Greece	Maya	11.2%		2.7%		3.5%	
Shared for all Wave 1	All wave 1 - Dog Products		14.3%	14.3%	12.5%	12.5%	10.6%	10.6%
Totals			100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Part count increase from Prior Study			Baseline		38.0%		-8.2%	

Table 3.12: Sharing analysis for the platform or underbody systems, first wave of derivatives, by region

				Unique Parts			Used in two Regions			Shared for all Wave 2	Wave 1		
				Greece	Egypt	Maya	Greece Egypt	Greece Maya	Egypt Maya	All Regions			
			WAVE UNIQUE	**9.58%**	**7.04%**	**3.32%**	**0.37%**	**2.76%**	**3.57%**	**3.29%**	Unique Wave 1	Shared w/ Wave 2	Total
Unique Parts	Egypt		**17.33%**	0.06%	0.71%	0.00%	0.03%	0.00%	0.16%	0.19%	17.33%	1.15%	18.48%
	Greece		**26.45%**	4.71%	0.00%	0.19%	0.03%	1.49%	0.00%	0.06%	26.45%	6.48%	32.93%
	Maya		**2.48%**	0.00%	0.00%	0.09%	0.00%	0.00%	0.06%	0.00%	2.48%	0.16%	2.64%
Used in 2 Regions	Egypt	Greece	**4.50%**	0.28%	0.06%	0.00%	0.25%	0.00%	0.06%	0.28%	4.50%	0.93%	5.43%
	Egypt	Maya	**0.50%**	0.03%	0.06%	0.03%	0.00%	0.00%	0.00%	0.00%	0.50%	0.12%	0.62%
	Greece	Maya	**1.43%**	0.87%	0.00%	0.06%	0.00%	0.03%	0.00%	0.12%	1.43%	1.09%	2.51%
Shared for all Wave 1	All Regions		**3.72%**	0.53%	0.12%	0.06%	0.28%	0.09%	0.00%	2.67%	3.72%	3.75%	7.47%
	Wave 2		Unique Wave 2	9.58%	7.04%	3.32%	0.37%	2.76%	3.57%	3.29%			29.92%
			Shared w/ Wave 1	6.48%	0.96%	0.43%	0.59%	1.61%	0.28%	3.32%		13.67%	
			Total	16.06%	8.00%	3.75%	0.96%	4.37%	3.84%	6.60%	56.40%		100.00%

Table 3.13: Wave 1 to wave 2 platform or underbody systems part sharing analysis

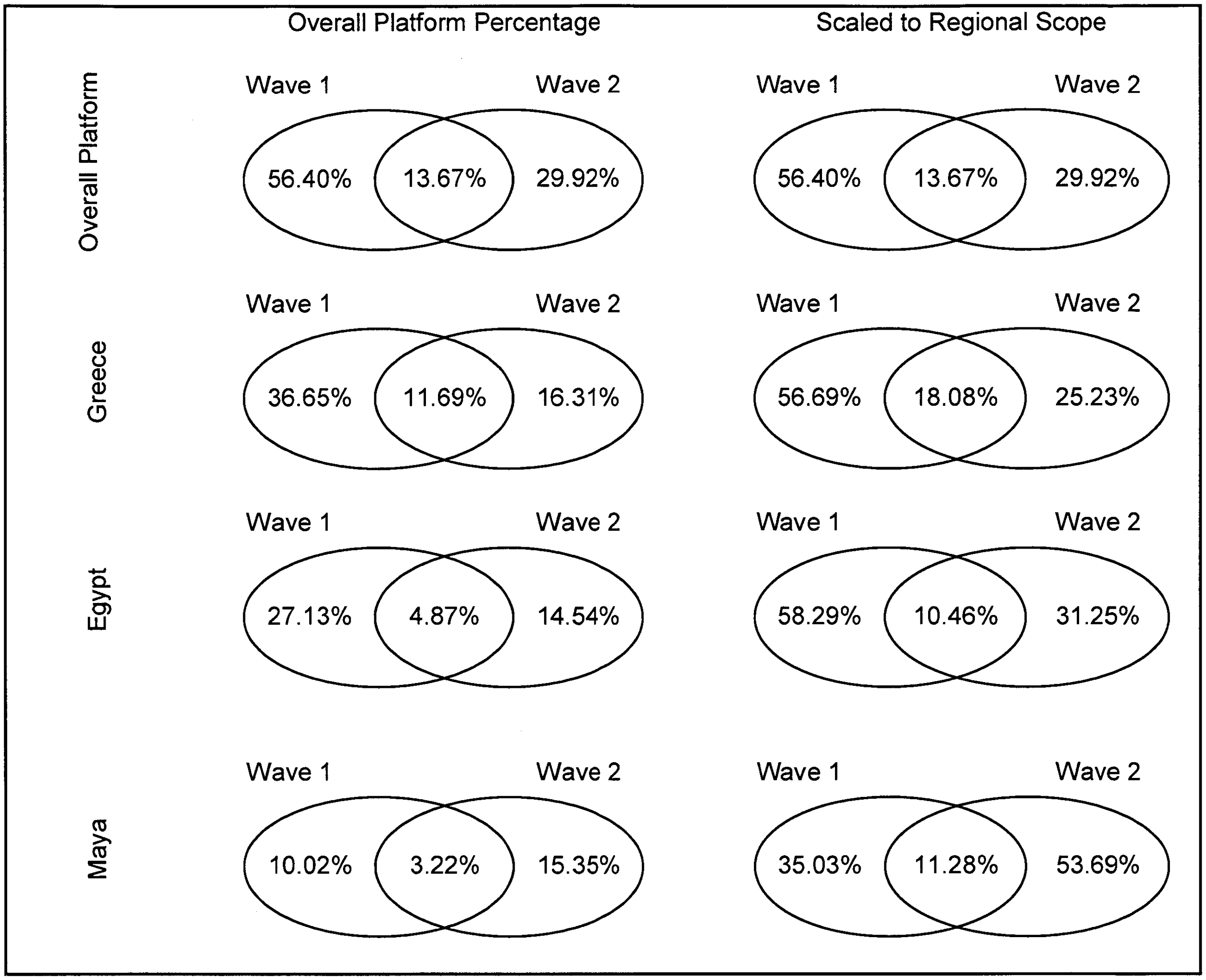


Figure 3.18: Wave 1 to Wave 2 Regional Sharing for the underbody or platform systems.

From all the data concerning the underbody or platform systems, the following can be concluded:

- Divergence is happening in both the Top Hat and the platform project: the total number of shared parts decreases over time and the number of unique parts increases.
- Surprisingly, for the first wave, the total number of shared parts for the underbody is lower, and the number of unique parts is higher (table 3.10 compared to table 3.6, or table 3.12 compared to table 3.8). The total complexity (total number of part variants) for the underbody is higher than the upperbody, this can be driven by the relative high number of available engines and transmissions and functional "burden" carried by the underbody.
- For the second wave (table 3.11 compared to table 3.7), the underbody systems had a higher commonality than the overall vehicle, as expected. Nevertheless, the relative part sharing is just marginally higher than the overall vehicle.
- If taking into consideration the parts that are shared with the second wave, the relative efficiencies of the product platform are noticeable. After comparing figure 3.17 to 3.16 and table 3.13 to table 3.9, the relative number of shared underbody parts compared to the overall vehicle almost doubles. In the specific case of Greece, 18% of the underbody parts for both the first wave and second wave are shared.
- The product family plan has not changed significantly over time and a new project organizational structure with modified processes have been settled, even though the relative amount of shared parts can be considered low.

When the sharing data was presented to the platform team, they were surprised about the relatively low number of shared parts between projects and the relative high number of unique parts for all the different projects, from both a Top Hat and a platform perspective. Having a relatively high number of unique part numbers drives more work (and rework) for all the different areas:

- Design and development: every part requires a blue print and design maintenance. In many cases, even if the parts are unique, the change can impact several part variants.
- Vehicle integration: more prototypes are required to build all the different available combinations, thus necessitating additional validation. It is more difficult to assess if a problem can be isolated to a specific build combination or affects several build combinations
- Project management and project integration: more part numbers imply longer BoMs and therefore require more time to finalize integration assessments.
- Purchasing: more part numbers will require more purchase orders for production and prototype parts.
- Finance: will require more time to assess the average cost of the vehicle.
- Manufacturing: will require other tools to effectively install the correct part number in each of the vehicles. Examples are scanning tools or sequencing parts.

The relatively low number of shared parts (and therefore, a high number of unique parts) can be attributed to the original sharing assumptions and the design commonality divergence throughout the project. The following section will combine the effect of the project and team structure and the project governance and the effects it had on product commonality.

3.6. Product Commonality Divergence and Convergence

3.6.1. Platform Change Control and Product Decisions per Product Development Phase

The commonality information presented in the prior section reflects a summary for all the different product design decisions that drove commonality divergence in the product. However, all these design decisions have to be further investigated to understand the reasons for this divergence. According to Boas (2008), the following are potential sources for divergence and their enablers (figure 3.19):

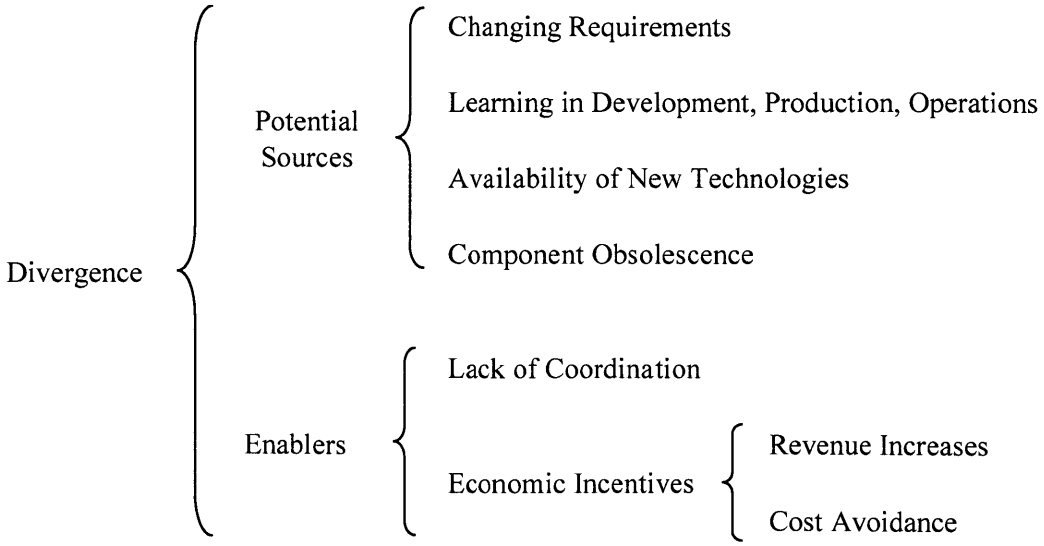


Figure 3.19: Potential sources and enablers for product commonality divergence. Reproduced from Boas (2008)

The platform change control meeting was the major tradeoff forum where the affected activities brought their design issues to define the best design solution with the platform project manager. To identify the details behind the product decisions that drove divergence, the platform change control meeting minutes were deeply analyzed to better understand the individual design decisions and the reasons that supported those decisions. In general, if a functional engineering area required a significant design change that drove either a significantly higher cost or a different overall product performance, the project governance required them to bring their changes into the platform change control meeting to concur with the best design decision and to cascade these decisions to all the affected activities through the meeting minutes. These decisions will be explained in the context of the three main development phases as outlined in figure 3.1: concept development, underbody detailed design, and underbody validation.

3.6.1.1. Conceptual Development Phase

During the conceptual design phase for the first wave of products, the apparent product commonality as measured by the company with the C, M, N and U classifications suffered the largest drop from phase to phase: 24.4% of the carryover components for the overall product became new or modified, representing a 19.8% drop for the platform subsystems (according to figures 3.11 and 3.13). The first phase represents the hand off of the project from the planning organization to the new project organization and its project managers that would execute the first wave of the product family plan.

The main design workstream for this first conceptual design phase of the project is the development of engineering concepts and alternatives in order to execute the product vision, technical specifications and requirements as outlined in the planning phase. These activities are now performed by the engineering functional areas in a bottom up perspective instead of the relatively small development team in the planning phase that developed the overall product strategy in a top down view. The role of the project manager is to reconcile these two perspectives and select the adequate engineering concepts that will be able to execute the product vision within the cost and functional targets.

The change of perspective, planning vs. execution, and top-down vs. bottom-up, presented some disconnects with the carryover content. The bottom up perspective represented the proposed designs in order to execute not only the product plan, but also the functional development plan, which is owned by each of the functional areas. This plan detailed the design and sourcing strategies for similar parts (also named commodities) for all the different vehicles produced by the company in order to support the required functional performance and reduce the variable cost. These relatively new commodity plans drove a significant change in the electrical architecture (electrical modules and wiring system), new lower cost driveshafts, cooling fans and exhaust systems, new plastic pedals, and a new air induction system (increased functionality) that were originally planned to be carryover.

In addition to the execution of the functional commodity plans, the planned product functional attributes also required some other changes that were not originally assumed in the planning portion of the project. Perhaps the most important functional attribute that drove most of the changes was the overall product weight reduction and improved fuel economy. Between these changes, a new common fuel tank (new tank volume) was required and also several parts changed their material from steel to aluminum to reduce the weight of the vehicle. Other functional attributes that drove changes were the enhanced safety performance and interior roominess that drove changes to the front and rear floor that were originally assumed carryover.

Another stream of changes not considered in the commonality plan were the introduction of new technologies that will add value to the customer, including a new computer controlled suspension system for the luxury segment; new fuel economy technologies, and more electrically instead of mechanically controlled systems. These product changes driven by new technologies were not originally envisioned due to the relatively broad scope of the planning organization instead of the focused knowledge of the functional engineering areas that are continuously benchmarking their designs with the competition and looking for opportunities to refresh the architecture of their systems for better performance and cost. As these changes were beneficial to the customer and the project, they were included into the scope of the product.

On the other side, the hand-off from the planning to the engineering functions also drove divergence due to product clarifications on the original assumptions. The more detailed analysis coming from the engineering functions resulted in additional new content that was originally carryover; the most significant changes were the suspension system and the braking system, which were optimized to the latest engineering assumptions, driven mainly by the planned vehicle weight. Also, in a similar bottom up vs. top down perspective, the commonality assessment process was lead by the engineering functions with the product's BoM rather than the top down assessment with surrogate costs that was performed before the project officially started. Before the project is handed to the development team, significant manual inputs are required in order to have the preliminary product commonality assessment. These different processes are detailed in figure 3.20. These process differences also drive apparent divergence in product commonality, as assessed in the company.

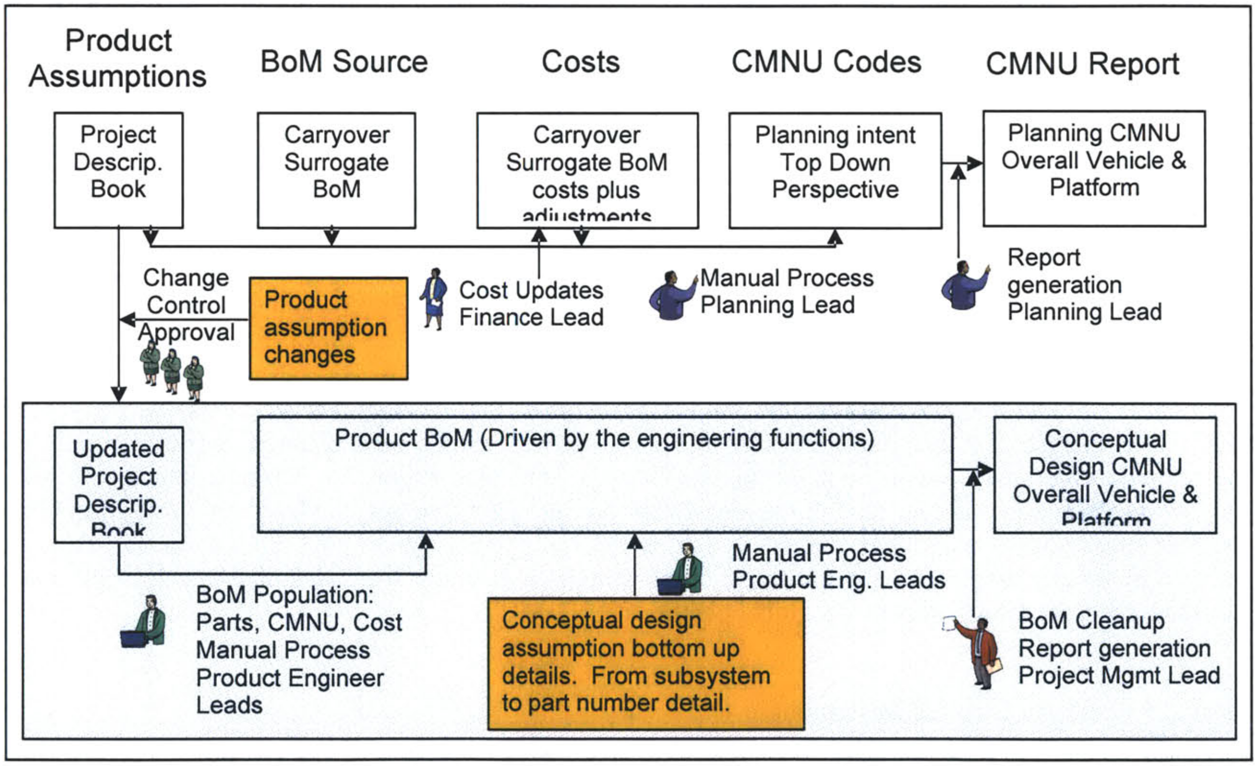


Figure 3.20: Company's process to generate the commonality report before the project starts (top) by the planning organization and after the project starts (bottom, in green) by product engineering and project management. Several manual inputs and surrogate assumptions are used for the original commonality report.

The vehicle integration function provided significant input to the conceptual development phase by validating and trading off all the different subsystem concepts and making sure at a vehicle level they were consistent with the overall vehicle performance attributes. As previously stated, the vehicle weight took a more relevant role in the development of the engineering concepts; however, the weight needed to be traded off with vehicle safety (crash) and noise vibration and harshness (NVH) which are usually more robust with wider sheet metal gages. In general, all these vehicle attributes were studied considering only the first wave of vehicles and were usually not studied or traded off with the overall platform commonality with the subsequent platform waves.

Many examples of design divergence were found in the conceptual design phase; nonetheless, the platform design team evaluated other alternatives for converging designs. Another stream of relevant design decisions with significant impact to the platform design and platform bandwidth was the selection of the powertrain combinations (engine and transmission) for the derivatives. During this phase, the design team reevaluated the originally proposed strategy and proposed some changes for reducing the overall number of combinations, becoming the most important source of design convergence. Furthermore, in an even higher level of strategic decisions, the total number of derivatives was also reconsidered. As a result, two of the original derivatives were deleted and one was added, although, these derivatives did not represent a significant additional development effort or manufacturing complexity.

Most of the discussions were centered on the first wave of vehicles and most of these were focused on the first derivative (Egyptian Dog), given that all the other first wave derivatives would reuse the same architecture. Nevertheless, some conceptual design reviews incorporated the effect into other waves, such as the brakes sizing strategy that was planned for the complete platform bandwidth. Perhaps one of the most important architectural decisions that took advantage of using the product platform approach was the location of the fuel filler side; the best alternative was chosen after evaluating the impact to all three waves. However, a different decision would have been taken if consideration had been given to only the first wave of derivatives in isolation. These examples show some of the benefits of having a platform design team and a platform project manager, where the original assumption was reevaluated and improved. While the definition of the fuel side was the best decision, it represented a different position from the donor platform and represented another source of divergence, showing that divergence can also be beneficial. Unfortunately, most of the decisions were assessed considering the best alternative only for the first wave of vehicles, as the information presented was based on the first wave alone.

The intent of the conceptual design phase was to confirm the original product strategy and to achieve a compatible product and project after the project hand off to the development activities. Yet, given the significant changes for the platform detailed above, as well as the new planned product styling, the conceptual design phase became a major update to the product plan although it was now developed by the engineering functional areas. Due to the significant product plan changes, the learning curve from being the first time to develop a global product, and other external factors, the original production start date was delayed by five months; three of these five months were used to extend the conceptual development phase to achieve a more robust plan with a global perspective.

3.6.1.2. Underbody Detailed Design Phase

The intent of the underbody detailed design phase is the completion of all the underbody geometrical data validated by analytical and computational models, ensuring that all the different components are compatible in the mechanical package of the vehicle for the desired vehicle performance. The other key workstream happening in this phase is the selection of the supply base for all the components (upperbody and underbody). Towards the end of this phase, the upperbody selects the final product styling so the final geometrical details for the underbody can be corrected in order to have successful vehicle integration. During this phase, the carryover content for the lead project (Egypt) decreased only 2.1% for the overall vehicle, but most of the divergence was driven by the underbody that reduced by 6.4% the percentage of common parts per figures 3.11 and 3.13. These figures also show how all the other derivatives also lost commonality in all projects but the Retriever projects were most affected, where the commonality loss was even deeper than the prior 2.1%.

As part of the design progression from a conceptual design to a detailed geometry for the underbody and a defined styling for the upperbody, the bill of materials also progressed significantly to capture all the required parts to build the vehicle. During this phase, the official bill of materials for the project became fully functional and the complete design input was available from the functional engineering areas through the bill of materials. This was the first time the commonality assessment truly reflected the latest engineering assumptions for all the parts in the bill of materials, compared to the prior phase where a significant amount of surrogate parts were included.

Some important differences with the prior phase may have helped to decrease divergence as reported by the company compared to the detailed design phase: the stability of the design team (instead of the responsibilities transfer from one team to another), the relative impact of the sourcing plans to the project was not as important (given that most of the strategic decisions had been committed already) and in this case, the area of focus was expanded from the original most expensive components and systems to a more complete view for the overall vehicle with less expensive components. Regardless of the scope focus increase, divergence now happened in other vehicle variants and their respective part variations (cousin parts) which were required to support the different powertrains, drive configurations or other vehicle features. Divergence was present, but happening outside of the cost control model, which kept being the largest focus for aligning cost, weight and engineering performance attributes, especially with the Egyptian and Greek Dog projects.

The underbody detailed design phase (for the lead projects: Egyptian Dog and Retriever) had a significant overlap with the upperbody detailed design phase and a small overlap with the Greek Dog underbody design phase. Also, the end of this phase was also concurrent with the start of the concept development of the second wave of products. The larger number of concurrent design workstreams resulted in cautiously paying attention to the integration of the follow on projects, which were not the primary focus during the concept development phase. Specifically, the Greek Dog project received more attention during this second phase compared to the first, probably as much as the lead project. Also, the imminent start of the second wave of derivatives got more attention from the execution teams, even before the formal hand off from planning to execution of the project as captured in figure 3.7.

Given the nature of the detailed design activities, the divergence was mainly driven by the mechanical package integration for all the underbody components and CAE studies were performed with the available geometrical data which resulted in some unfeasible designs. Also, there was closer work with the supply base, which also raised some manufacturability issues that caused design churn. Examples of design divergence in this phase were: a new clutch because of an infeasible package with the new pedal; several front structure and floor stampings were redesigned from the carryover product driven by safety crash analysis and were either reinforced or changed in shape; new mounting brackets were found to be required but not initially assumed given the space constraints in the vehicle. This phase had a larger focus on production costs, so most of the design decisions were done trying to optimize it; even some of the prior conceptual designs in aluminum were reverted to steel to achieve a lower cost.

Closer work with the manufacturing activities (either vehicle assembly or components fabrication) also brought some divergence. The original intent was to use the same assembly sequence or bill of process from the donor platform, also considered part of the platform itself, which in most cases was followed for the design execution. However, as the designs were progressing, a few additional opportunities were identified to reduce the required investment for the new product, but they required the designs to be slightly different between derivatives. As an example, one of the underbody structure assemblies changed from welded to bolted since the new size did not fit the existing e-coat tank of the selected supplier.

These type of design details will never be available in the planning phase, but these situations enable divergence for positive economic reasons.

On the other hand, convergence opportunities were also pursued like commonizing major stampings for the different subframe variations, reverting to existing technologies due to high costs, reverting components to industry standards, and reducing the total number of alternators and batteries available. One of the most important commonization opportunities achieved was reducing the combination of executions of the rear fascia and the chromed exhaust tip, which would not have been achieved without a strong desire for commonization and reduction of the total number of exhaust assemblies by the platform project manager. Also, some imminent changes for the second wave were considered during this phase to avoid future derivatives divergence, an opportunity that would have never been possible with a conventional project structure without a strong platform project manager.

All the strategic decisions should have been decided by this phase; however, given the significant updates to the original product plan, more changes were required in order to achieve the forecasted production costs. In the first stream of strategic decisions, the total number of derivatives was increased as a new derivative was planned to improve the revenue of the project. This new derivative was intended to reuse the exterior styling and an existing powertrain. The second stream of strategic decisions was the re-evaluation of the existing powertrain offerings. These were the most important sources of design churn for the underbody:

- A different existing engine was added to the Egyptian Dog project
- One of the shared engines for all regions was changed for a completely new, smaller, and cheaper engine with a similar architecture. Because this was a late addition, it was decided not to introduce the new engine in the lead derivative, but planned to be introduced after the start of production.
- Changed a unique powertrain for Maya to a shared powertrain, causing design convergence for these derivatives.
- Agreed to offer additional emissions levels for less stringent regulations instead of sharing the same calibration for all markets.
- Some powertrains were deleted temporarily, but after re-evaluation they were added back to the product plan.

The net effect of these powertrain offerings was a lower average cost. Nevertheless, the total number of offerings grew and a new engine was added. This late addition would have never been considered in a normal development environment, however, given the considerable lifecycle offset between the Egyptian Dog and the Greek Dog projects, it was a clear opportunity to introduce the new engine into the plan.

A third relevant strategic decision that was considered in this phase was a major redesign of the rear suspension for all three waves of derivatives to achieve a lower cost. The modification included changes not only in the suspension system, but also in the rear floor and the exhaust system. The change was evaluated late into a normal underbody development, however, similar to the new engine, the lifecycle offset between the two projects created an opportunity to study and incorporate this design and adapt the lead project with the new rear suspension. This change is another example of beneficial divergence, which was enabled by the lifecycle offset between the projects.

The imminent project start of the second wave of derivatives caused a new stream of changes required for the underbody of these derivatives compared to the first wave. These changes were driven by the required larger tires and the increased gross vehicle weight and loads that were needed for the second wave. One of these changes was a relatively small lowering of the position of the engine and transmission, but it drove significant changes to the peripheral systems that had to be adapted such as the driveshafts, the air induction system, the exhaust, and the mounting system. When these changes were evaluated for incorporation into the first wave of derivatives to achieve a common design, it was decided not to do so because of the advanced phase of their design, causing the second wave to have unique designs even before their project started. At this phase, the second wave of derivatives was still led by the planning organization and was not approved (or at least required) for the engineering functions to start studying changes for this wave.

The underbody detailed design was driven by the optimization and the execution of the conceptual designs derived in the prior phase. The progression of the design usually reveals issues driven by packaging constraints, manufacturing limitations, and preliminary engineering model assessments (like CAE or CFD) that require the designs to be updated accordingly to achieve their required functionality. All these situations are very difficult to forecast during the planning phase, and therefore divergence happened in a relatively natural way as part of product development. Having the platform and Top Hat structure separate allowed for more focus on the design progression of the separate underbody and upperbody workstreams. However, it is impossible to quantify if the divergence in this phase would have been higher, equal, or lower without the organizational structure as these issues are common in product development even of single vehicles and are part of the normal design progression.

The project structure and the lifecycle offsets allowed the project to make rational decisions in pursuing or not pursuing alternatives given the impact to later derivatives. The introduction of a new engine and new suspension were conscious and positive decisions for the project, and they were made possible because of the lifecycle offsets and the leadership of team members in charge of executing all the derivatives, and not by considering one set of products in isolation. The only real obstacle to pursuing common designs is the required time needed to develop and validate a proposal that works for all products. A closer start date of production becomes a major hurdle for adapting the design of the lead derivatives. Regardless of the benefits of a common design, the changes can only be pursued if the design schedule allows redesign of the common parts without compromising the integrity of the product.

3.6.1.3. Underbody Validation / Upperbody Detailed Design (Wave 1) and Conceptual Design (Wave 2)

The third development phase consisted of several design phases which were overlapped within the different projects. For the lead project it comprised the underbody validation, however, different phases were also included as detailed in figure 3.21. This richness of overlapping projects became very beneficial as it allowed adjusting the designs in order to achieve product commonality. The lead vehicle was stable during this phase, losing only 0.1% commonality of the overall vehicle, even showing convergence of 0.9% for the underbody (driven mostly by cost updates instead of actual design convergence); however, most of the follow on projects, including the second wave, had significant divergence as shown in figures 3.12 to 3.15.

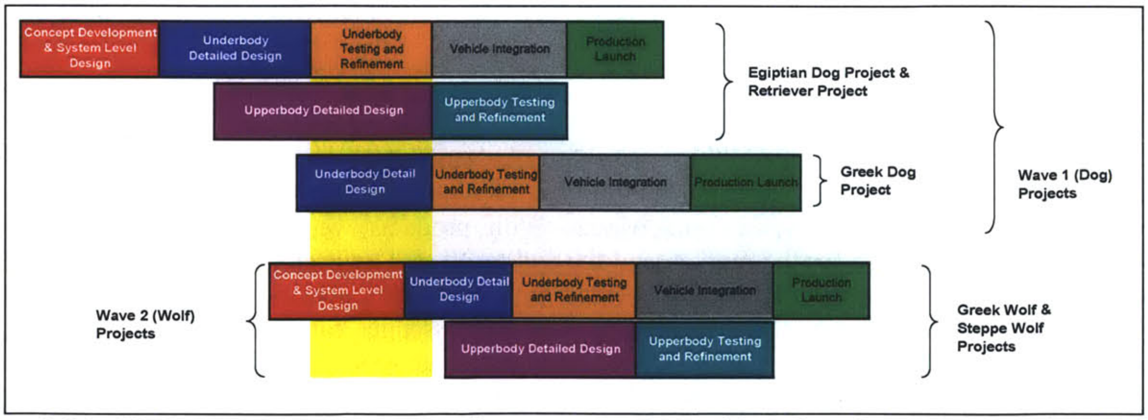


Figure 3.21: Underbody testing and refinement overlap with other development phases of the product family project.

During this period each of the phases has different purposes:

- Egyptian Dog project underbody – During the testing and refinement phase, the underbody prototype was built and tested. The intent of this phase is finding the design bugs so they can be fixed before the final production.
- Egyptian Dog upperbody – The upperbody detailed design phase is when all of the drawings are developed and integrated into the upperbody package. These geometries are inputs to the upperbody CAE studies that now validated the vehicle as a whole entity. The designs are progressed and optimized for cost and weight, now that vehicle styling has been defined.
- Egyptian Dog upperbody and underbody – Once the design of the complete vehicles has been completed, all these "final" versions are sent for final agreement to the suppliers, which should be signed-off by the end of this phase.
- Greek Dog underbody – Development and completion of the drawings and underbody package. The project decided to stagger the Greek Dog development because of the significant amount of unique parts because of unique engines and transmissions. The design and development of two new engines for the Greek market were perhaps the most important changes to the platform.
- Wave two (wolf-like) projects – Realization of the conceptual designs for both the upperbody and underbody, overall project evaluation for cost, and functional attributes compatibility. For the underbody, conceptual designs were developed only for the parts that needed to be altered due to new requirements such as a new engine development, increased passenger capacity or new braking performance. Most of the underbody conceptual designs were nearly the same designs as in the first wave, at least in an architectural perspective.

For the first wave of vehicles, this phase was characterized by relatively stable strategic decisions that drove some convergence: neither engines nor transmissions changed during this period and one Top Hat was deleted from the plan. This Top Hat had a slightly different exterior look and was effectively deleted from the platform and a significant investment savings was achieved as a result. However, no platform components were affected by this deletion, and the name of this project/product (the Dachshund) was never included in the platform plan described in table 3.1.

The second wave of vehicles also offered a relatively stable powertrain offering strategy during this phase, similar to the first wave. The original plan reused three of their four engines and only required the development of one new engine; however, towards the end of the conceptual design phase, another engine was introduced to the plan for the Tundra Wolf, with the main purpose of differentiating the Egyptian Wolf and the Tundra Wolf products but also to achieve lower costs. This added engine was also reused from the first wave, meaning the second wave would offer five engines, and four would be carryover. The divergence effect of this change was not captured in the data above because it was one of the latest decisions towards the completion of the design phase.

In regards to the Top Hat offerings, the second wave had many more concerns with the exterior and interior styling commonization than the first wave. The original intent of the second wave was to develop a global product, however, after market research, the plan was deemed infeasible and therefore caused significant churn in the products' direction before final styling was selected. Originally, prior to the project start, all the Greek, Egyptian and Mayan Wolf products were all alike. Later, a minor modification was not enough for the market, and the end result was a different styling for all three. The Greek styling was more conservative and concerned with passenger capacity, the Egyptian styling was required to be more aggressive without the need of extra passenger capacity, and the Mayan market wanted a mix of both. The end result of this major strategy change was the need for more unique parts for each product; and, therefore, the investment and development costs increased. Nevertheless, the platform components remained relatively stable since the platform extent or bandwidth was carefully planned and only required few modifications in the rear floor for each product, which was already considered new for the second wave. Comparing tables 3.13 and 3.15, the overall vehicle carryover content decreased. Although, the commonality decrease was larger for the complete vehicle than the platform specific content, meaning that divergence was driven by the number of unique upperbody components. On the other hand, one convergence opportunity was executed (a common front end), and one additional tophat for a different region was deleted from the original plan (the "Persian Wolf"), which also included a new plant for its production. This derivative was never included in table 3.1.

During this phase, divergence for the first wave of vehicles was more evident in other product variants than the selected control models and divergence was even more evident in the follow on projects. The evidence of divergence happening outside the control models is shown in the sharing studies (tables 3.7 to 3.13) that were performed during this phase of the project when the official bill of materials was functional and available for both waves. Discipline to learn and use the official BoM with full functionality was much easier for the second wave than the first wave, revealing another advantage of the product family project organization.

For the first wave lead vehicles (Egyptian Dog and Retriever), several changes were required due to the design bugs found during testing. Nevertheless, most of the affected parts were already assumed new so the changes did not represent significant divergence. Changes were required for various reasons such as enhanced safety, better vehicle dynamics and functional failures, therefore new parts were designed for all the platform derivatives (shear brackets, additional mass dampers, new powertrain mounts, unique transmission case for Retriever). Some failures were found on specific vehicles and design updates were incorporated in only the vehicles where the failure was found, causing divergence between variants and parts proliferation that would avoid bigger average costs. The overall philosophy of the project manager was to fix only the required vehicles in order not to penalize all the derivatives, unless evidence was strong enough to incorporate these changes in all the products. Examples of these changes were a new oil pan for one of the engines, alterations required to handle fuel with larger ethanol content in specific markets, different pulleys for similar engines, and a larger brake booster for one of the engines (already planned for the second wave).

In the case of the first wave follow on vehicles (Greek and Mayan Dog), divergence was mainly caused by the progression of the underbody packaging of the new engines and the related CAE and CFD analysis. Since the only differences between the Greek and Mayan Dog vehicles and the lead vehicles were dissimilar engines and transmissions, all the other underbody components such as the suspension, steering, brakes or the floor pan were kept stable, as these remained common between these derivatives. Given that the design changes were caused by different engines and transmissions, as they were already unique to these vehicles, it did not cause significant divergence. However, the parts proliferation found in the parts sharing studies show this divergence as many of the parts were originally assumed carryover (perhaps for simplicity) or did not exist in the BoM. Specific examples of this divergence were new air induction ducts and new halfshafts for the recently added engine (instead of carryover).

Other sources of divergence in this phase were the execution of the sourcing plans and the divergence of the donor platform. Alternative suppliers were sought and some of them offered lower production costs, but the lower costs were achieved with a new design instead of the common design. These suppliers were usually preferred given the economic advantage, thus causing incremental divergence. On the other hand, the donor platform that was in production continued to have design changes, and in some cases these updates created changes to the surrounding parts or even a new design. Both these sourcing and donor platform changes could cause either divergence or convergence. In addition, while these changes were found in this specific phase, the sourcing and donor platform changes could be found in other phases too.

As the upperbody parts continued their development, some issues were found when integrating them into the already designed platform. In many of the cases, it was easier and cheaper to accommodate the required changes into the platform components instead of over-engineering the upperbody components or designing bridging parts for their integration such as support brackets. These changes due to the later development of the upperbody also drove divergence to parts such as the wiring harnesses, the cross car beam, and the front structure that were adapted to the changing upperbody parts designs. Also, once the complete vehicle was designed, additional CAE studies show the need for additional reinforcements in the floor (to avoid NVH issues) which themselves increased the weight of the vehicle. To achieve the vehicle weight, some of the once reverted steel components changed back to the original aluminum design, increasing the design churn; however, as these were new already, this did not represent divergence.

Once the second wave of derivatives project was transferred from the planning team to the executing team, the project and the product required some changes to the original plan. Some of the design assumptions in the commonality and differentiation plan were found not to be feasible by the design engineers. Besides the strategic styling changes, the updated powertrain offering, and the deletion of one of the derivatives, some other non-strategic changes were required to achieve product functionality resulting in design divergence. Some of the changes that caused divergence included the development of a set of bigger brakes for some of the more demanding applications (which later required to be used in a larger number of derivatives), the development of a modified HVAC case, additional changes to the cross car beam changes; and larger condensers, radiator, and fans. These parts were planned to be reused from the first wave, however, the more demanding application drove new designs.

The introduction of the new engine for the products in the Egypt region drove incremental changes to other systems such as a new starter or a modification in the cooling system (not previously assumed). Nonetheless, they were not captured in the corporate commonality metrics as they were not part of the control models. The second significant difference in Egypt was the addition of a new assembly plant, however, the new plant did not drive additional divergence, since it was planned to be updated to use the same bill of process as the other derivatives. In this case, the high willingness to commonize designs resulted in a greater investment in the supply base and the manufacturing location in order to handle the new bill of process.

Besides the actual product changes that resulted in divergence for the second wave, the effect of the project handoff explained in figure 3.20 was also relevant in the apparent divergence as measured in the company. Divergence was caused by cost changes, and further clarification of the design whether the part was carryover (shared with the first wave), modified (cousin part) or new. No new technologies were introduced from the start of the project to the conceptual development phase. The impact of weight reductions, NVH improvements or safety performance improvements was also negligible since many of these changes were already in place for the first wave and the second wave would follow a similar product performance, only with added capacity for higher loads, additional passengers and changes required for additional climate control performance. The second wave also did not have divergence driven by the functional design and sourcing plans. The overall result of all these situations was a lower divergence from the project start to the end of the conceptual development phase for the second wave (carryover parts reduced from 54% to 41%, figure 3.13), compared to the first wave (53% to 29%, figure 3.12).

The conceptual development phase represented the first time that the platform project manager and the development team were both engaged with developing derivatives for both waves. This overlapping of projects and teams was quite beneficial for the products, since it enabled the desired interaction for developing product platforms. The same team was accountable for both waves and sensitive to all design decisions, and the cost, performance and commonality tradeoffs were handled better than in the prior phases. This "overall picture" understanding and accountability enabled improved decisions and also facilitated design convergence. This overlap allowed for either tradeoff discussions for both waves concurrently or new design updates for the second wave, given the learning in the first wave as a "catch up" process. The most relevant example of design convergence was the early introduction of the second wave changes required to the front structure, enabling a common front structure between the two waves. The benefit of a common assembly process exceeded the burden of a more robust and costly structure for the first wave. Another key convergence action was enabling a common wiring routing between the engine compartment and the interior of the vehicle, which found significant packaging issues with the second wave of derivatives. Finally, other relevant discussions for commonality tradeoffs were regarding the required towing performance, and the selection of the transmissions, drive ratios, radiators and fans to achieve the required performance. While the first wave had discussed these issues, they had to be revisited once updated information from the second wave was available. The selection was finally traded off between both waves in terms of cost and performance.

Team interaction not only enabled convergence, but also allowed for improved discussions on the impact of design decisions. While commonality was desired, some convergence opportunities were not pursued given the relatively advanced design of the first wave of derivatives compared to the second. The risk of changing the design in such an advanced phase outweighed all the other improvements. This was the case of the intent to communize one of the rear floor crossmembers, where the required updates to the seat structure and fuel tank for the first wave of derivatives were assessed as infeasible given the development schedule constraints to the delivery of prototypes and the start of production.

The existence of the platform project manager allowed for guidance of the product decisions and higher commonality, either by trading off commonality with other functional attributes, cost or schedule risk or by pushing the teams to find a common solution for all the derivatives. The platform project manager would not allow divergence if there was not strong evidence that it was the best design solution. This mindset allowed revisiting some unique designs like a voltage stabilizer that was originally assumed different between the markets, but finally a common solution was found; or, by not allowing unique designs even if they had great benefits. For example, a proposal for a different material instrument panel structure was rejected since it would require changes to many parts and drive unique assembly processes.

Another phenomenon that was reproduced again in this phase was the relative loss of sight of the third wave of derivatives. There were few discussions that involved the second or the third wave during the conceptual development phase of the first wave, and again, there were few discussions involving the third wave overall, as shown in figure 3.8. The figure shows again the strong effect of the planning organization taking the lead of the project before it formally starts. Few discussions involved the second and third wave and only two of them were relevant. First, a new starter motor was required for the third wave and it was agreed to pull ahead the starter motor to the second wave. Second, a new power supply technology was also proposed for the second wave, and it achieved equal costs with enhanced performance, but given the risk for the second wave (even being in the conceptual design phase) it was decided to defer its introduction to the third wave of derivatives and studying a "carry back" solution, that is, implementing it in a future design phase or after production starts. Throughout the case study, not a single carry back solution was agreed upon or even studied for its incorporation given the high redesign costs for second wave derivatives; showing that once the opportunity of being common is lost, then it is very hard to recover it.

While design convergence was one of the most important contributions to the concurrent design of all the derivatives, it had some hurdles for its execution. First, given the lifecycle offsets of the projects, and perhaps the fact that they were driven as different projects instead of a single mega-project, there were project accounting difficulties. The challenge was found when a part that originally was identified to be required for the second wave of derivatives but was later found to also be required for the first derivatives as well; for example, the front structure commonization or the need for a larger brake booster for one of the derivatives. Given that the funds for developing and producing these parts were available, the same engineering team designed the part, and even the project manager directing the team to incorporate the changes into the first wave to make them common, the accounting between the projects was still difficult to implement, thus slowing the performance of the team.

This period was the most insightful time to understand the nature and the complexity of platform product development once the complete scope of the product family was managed. Regardless of the development projects that were going through different design phases, the overlap of the projects allowed for more interaction for its development even with the existing lifecycle offset. The nature of the decisions was more robust given the latest design information available from the lead project, which allowed having preliminary insights for the following derivatives or having real time discussions considering both waves to find the optimal solution for the product family. This insight further confirms Boas' (2008) proposed optimal lifecycle offset, as the case study was developed with a near to optimal lifecycle offset: large enough to allow resources distribution and short enough to allow the projects to correct the design based on the insights from both projects.

Unfortunately for the projects, approval of their design phases (as planned in figure 3.1) was deferred due to both higher overall vehicle costs and increased investments. The commonality data shown throughout the case study was cut-off at the original approval dates and does not reflect further changes before the actual approval, which occurred two months later for the wave 1, and was deferred six months for wave 2.

3.6.1.4. Summary

The three sections above described chronologically the progress of each of the three first design phases of the project. The intent was to provide some actual examples of the divergence found in the project and the drivers and enablers for them. Through this exercise, the framework described by Boas (2008) in figure 3.19 is now expanded with the specific input of the product development process, where divergence is now intended to be explained in the context of time and the PDP. Through this case study one of the most relevant findings is the impact of time on divergence, beyond the simple understanding of the lifecycle offsets. This framework is summarized in figure 3.22.

As shown in the figure, the divergence sources and enablers are further characterized in the context of the product development process. The intent of this framework expansion is to understand, at least conceptually, what divergence occurs naturally in the product development process, what divergence is driven by the changing requirements that could potentially be avoided, and what divergence is influenced by the lack of coordination among the team or economic incentives. For this means, the category of "learning in Product Development and Operations" has been selected as the natural source of divergence since this learning happens when information is received and progresses through the development of the concept, the design completion and the testing of these designs. This is visually illustrated in figure 3.21 where it is clearly demonstrated that in the case study the divergence caused by learning is dependent on the product development phase. On the other hand, all other divergence sources or enablers can be found in different PDP phases.

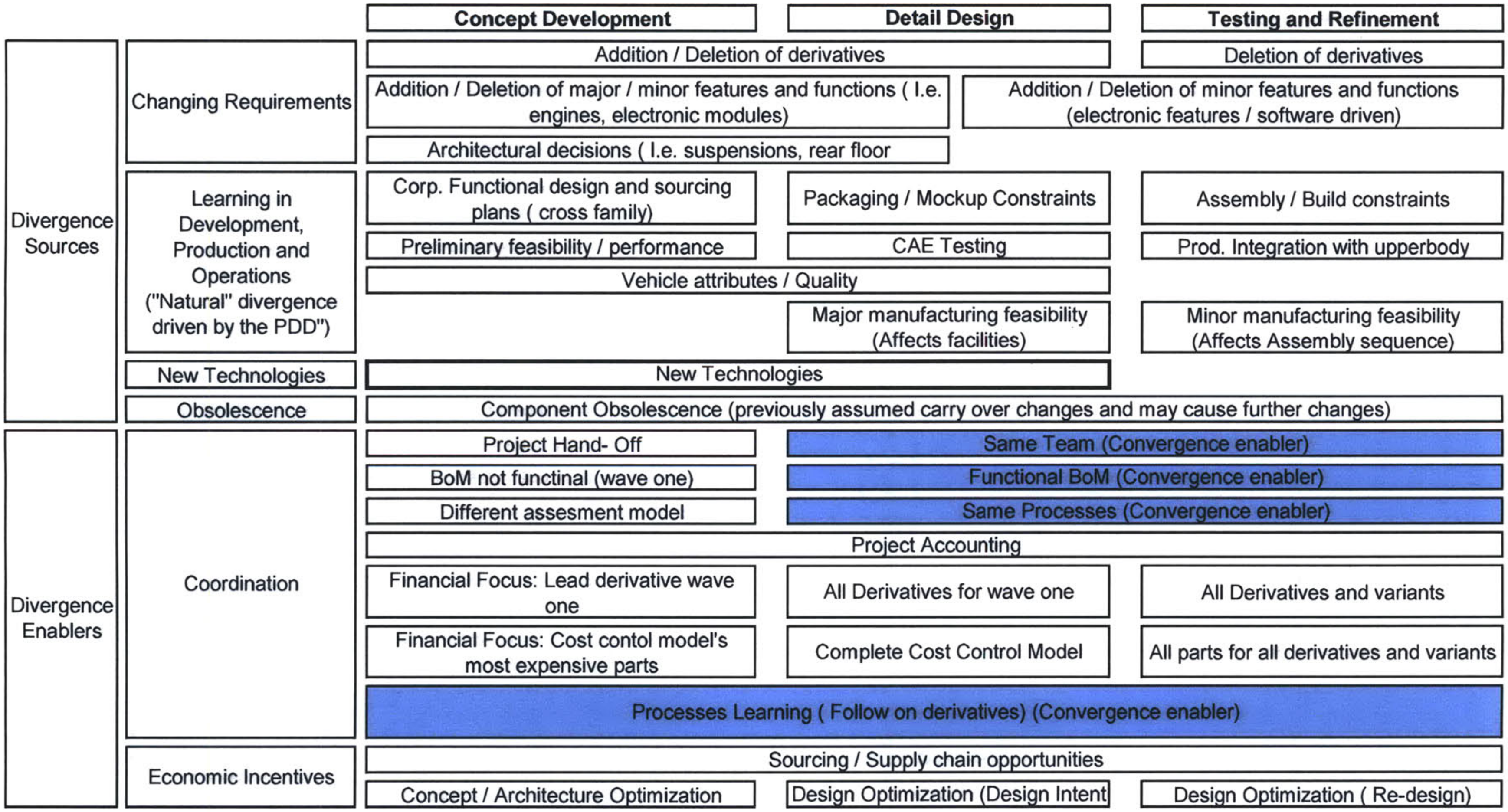


Figure 3.22: Divergence sources and enablers expanded framework considering PDD phases. The white boxes represents a factor for divergence and the blue boxes represent factors for convergence.

Achieving common designs is also significantly driven by time constraints. At the beginning of the project, commonality is the highest, showing an optimistic view of product planning which includes a simplified approach to share parts if there is not strong evidence for the part to be redesigned or requiring to be unique. At this phase prior to starting the project, it is feasible to change any part as there is enough time to design and test it; however, as the project progresses, the ability to redesign parts is lost. This ability to realize and plan relatively complex or strategic changes is shown in the "changing requirements" section, where a timing risk assessment should be balanced prior to deciding to pursue changes. The ability to change the product is context specific, but always time constrained. In the case study, decisions were made both ways, some changes accepted to be included, even if they were late (like the late addition of a new engine); however, other changes, like the crossmember commonization, were not accepted.

Figure 3.22 also includes a more complete understanding of some situations that caused coordination issues. It was found that the first months of development portrayed the highest commonality losses using the company's assessment method (C, M, N, U); clearly, it was during this phase where more enablers were found, including the project hand-off or the functionality of the corporate BoM. Some coordination situations can also enable convergence or lower divergence, which are shaded blue in the figure.

Another observation was regarding the coordination concerns, and how it is related to the relative attention given to the complete project scope. This attention can be divided into two main categories, the first one is related to the attention given to specific parts and the second related to the attention given to specific waves, derivatives and sub-variants. The relative attention to parts increases over time; an automobile can have around 10,000 parts in its BoM (Ulrich, Eppinger, 2008). Since it is impossible to integrate such a number of conceptual designs into the product, a Pareto rule is followed and the parts and modules that represent 80% of the value are identified first and managed through the conceptual development phase. Increased attention is given to their design and manufacturing source selection, these were called "key parts." Throughout the process, it was found that other parts required some level of attention and special (more relaxed) processes were developed to design them and select their supply base. Figure 3.23 shows the total number of quotation packages that were identified to be required for those parts outside the "key parts," a total which grew continuously. Those parts represented less expensive parts which some of them were originally planned to be carryover.

In the case of the relative attention given to the waves, derivatives and sub-variants also increased throughout the development. Table 3.14 represents this evolution throughout the different product development phases under study. At the beginning of the project, attention was centered on getting the right conceptual designs and most of the attention was given to the lead derivative. This increased attention could have been driven by the existing culture for designing products for only one market instead of considering the needs for global products. During the following phase, attention was almost equal between the two more important projects (Egyptian Dog and Greek Dog), however, the complete scope of the platform was still missing. Finally, in the last phase, the complete project scope for the first wave was studied, including all the variants for all the derivatives. In the case of the second wave, the positive learning from the first wave, the increased awareness for global products, and commonality with the first wave also helped to increase significantly the awareness of the complete project scope compared to the first wave, perhaps as complete as the detailed design phase in the first wave. The last row of the table shows an estimate of the relative attention to the total project parts as a percentage, verifying how the lack of attention to the overall project scope was also an important contributor to divergence. In other similar product family developments, a significant amount of attention is given to the lead project; however, in order to avoid detrimental divergence, all the derivatives should be carefully attended.

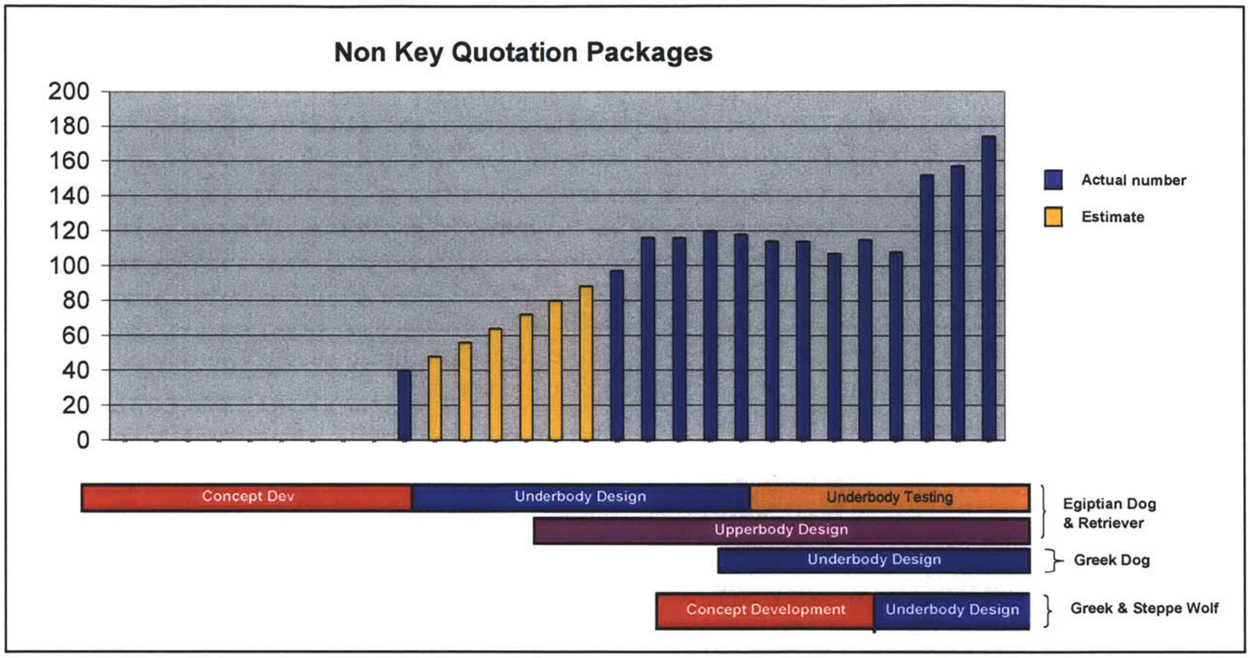


Figure 3.23: Increase of the number of quotation packages required for all the derivatives of the first wave of products. The bars in orange represent estimates.

<table>
<tr><th></th><th>Concept Development</th><th>Detail Design (Underbody)</th><th colspan="2">Testing and Refinement (Wave 1 Underbody)
Detail Design (Wave 1 Upperbody)
Testing and Refinement (Wave 1 Underbody)</th></tr>
<tr><td>Waves</td><td>Focus on Wave 1
Few Cross Waves Discussions</td><td>Focus on Wave 1
Some cross wave discussions toward the end of the phase</td><td colspan="2">Relative Focus on Wave 1 - 60%
Relative focus on Wave 2 - 40%</td></tr>
<tr><td>Derivatives Relative attention</td><td>ED 70%
GD 12.5%
MD 5%
R 12.5%</td><td>ED 40%
GD 25%
MD 10%
R 25%</td><td>ED 40%
GD 25%
MD 10%
R 25%</td><td>GW 35%
EW 30%
MW 5%
SW 5%
TW 15%</td></tr>
<tr><td>Variants</td><td>Cost control model only</td><td>Cost control model only</td><td colspan="2">Complete project scope with all variants for Wave 1 (All bodystyles, all products, all powertrains and all options and features)
Cost control model only (Wave 2)</td></tr>
<tr><td>Estimated % of parts under scope</td><td>20% of Wave 1 parts</td><td>40% of Wave 1 parts</td><td colspan="2">100% of Wave 1 parts
40% of Wave 2 parts</td></tr>
</table>

Table 3.14: Relative attention to the different waves, derivatives and variants.

The context of this case study emphasized the relevance of platform product development. A leader (platform project manager) was appointed and a whole new structure was in place to support the development: different processes and design progression for underbody and upperbody, a clear and relatively stable product family plan, teams assigned to support the new processes and organizational structure. Nevertheless, divergence still happened in all the different product development phases under study. Commonality declines over time in the context of complex product families as found in industrial practice (Boas, 2008). However, the PDP itself and the design progression are also intrinsic sources of divergence since even with strong commitment to not allow divergence to happen (as emphasized in this case study), the information will never be able to be complete, accurate, and timely to provide complete commonality direction before the development project starts. Design decisions are a day-to-day activity, and the best decision and tradeoff are always decided based on the available information. The risk of reducing commonality should be accounted for from the beginning of the project.

3.6.2. Divergence enablers: Coordination Concerns and Economic Incentives

The prior section showed specific examples of design divergence throughout the project phases and reasons for its occurrence. Sometimes divergence was caused by strategic and conscious decisions (changing requirements), but at other times it was caused by the normal progression of the product development process, calling it "natural" divergence because it was not enabled by a lack of coordination or by economic incentives or driven by a changing requirement. The intent of this section is to further understand the actions and decisions that the platform project manager can take to balance his/her "project iron triangle."

3.6.2.1. Process Related Coordination Concerns

The previous section identified some of the preliminary situations that enabled divergence due to a lack of coordination:

- Handing off the project from one organization to another;
- Not having a fully functional BoM in the early phases;
- Assessing commonality with different models or surrogate information, and making overly optimistic or simplistic commonality assumptions;
- Issues with the projects accounting that discouraged or did not reward the design engineers for achieving commonality between waves or even between regions;
- Limited scope of the total number of parts included in the integration of the project; and,
- Limited scope for the complete extent of the product family: waves, derivatives and sub-variants.

All of these situations could be avoided or minimized if the processes and tools in place were more complete and robust to fully and adequately assess product commonality at an earlier time. To avoid the situations above, the required fixes seem easy: achieve a better project hand off (or avoid the "throw it over the wall" situation), have a more functional and robust BoM earlier in the process, use actual data aligned to the vision of the executor of the design instead of surrogate information, and assess commonality for the complete scope of parts, derivatives and variants from the project start. However, executing these recommendations are not trivial tasks.

All of the recommendations above require an earlier existence of a larger team, even before the project starts, and may cause higher costs to the project. The inherent tradeoff is: does the risk of not achieving higher commonality outweigh the cost of having a "complete" team earlier in the project? Another key question is if having more resources upfront will make sense, given the uncertainty of the design and the unavailability of relevant information in the early phases. Usually, if engineering has not received the information required from other activities, having them work early in the project is a waste. There is a limit to the tasks that can be pulled ahead in the design. To assess if a part is going to be common or not, depends on how the design is going to be executed, given that information is usually not available earlier in the project. Therefore, any recommendation which implies pulling ahead workforce requires careful evaluation; the design structure matrix (DSM) (Eppinger et al, 2004) can be a valuable tool to these means.

3.6.2.2. People and Organizational Coordination Concerns

Having a capable workforce early in the project can help overcome some of the process related coordination concerns stated above; however, this workforce should also be organized efficiently in such a way that coordination between the different projects is emphasized to enable commonality. Product platforms' key development problem is finding the right mix of commonization and differentiation. If a part or system is positively selected to be differentiated, then assigning a different engineer for its design can be an effective way for achieving this differentiation; on the other hand, if the part is intended to be common, it makes sense to assign the same engineer to develop a common solution for all the different derivatives. This follows a simplistic line of thought: "Common parts – common engineer, different parts – different engineers."

The project followed a similar workload assignment strategy for the design engineers as stated above. Whenever it made sense to have one engineer responsible for different derivatives (given the high degree of commonality between the derivatives) he/she was assigned to all of them regardless if they shared the exact same design or they were cousin parts. Figure 3.24 below details how the engineers and their supervisors were assigned to each of the projects (waves).

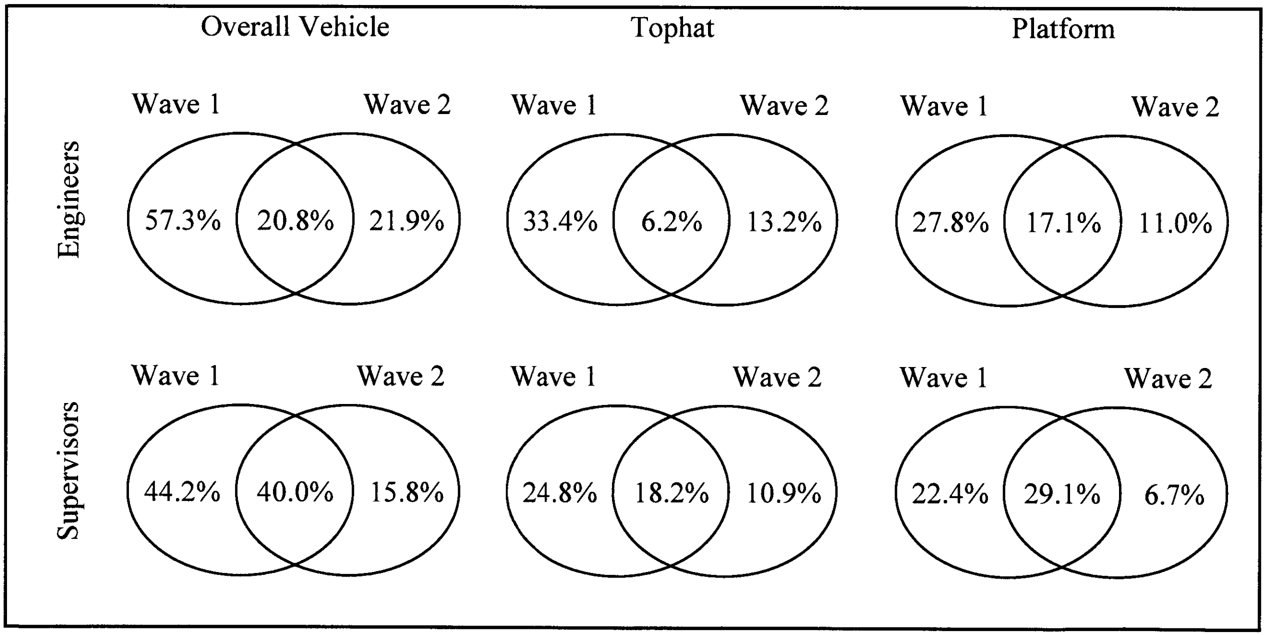


Figure 3.24: Number of engineers and supervisors assigned to the wave 1 projects, wave 2 projects or both. The first figures at the left represent 100% of the workforce and the following two figures (Tophat and Platform) represent the breakdown for each.

Some observations can be derived from figure 3.24:

- The assignment to a Top Hat or platform project is not clear for some engineers. They may develop multiple parts: some of them were part of the platform project and belonged to the Top Hat project (representing around 8% of the total number of engineers and 11% of the engineering supervisors). This can cause confusion in the engineering departments to identify which project manager should approve their designs and changes.
- The workforce assigned to the first project is larger than the second. This is a common practice since peak resources are required during the detailed design phase, and fewer resources are required during concept development.

- Engineers are shared to a higher degree for the platform components than the Top Hat components, as expected; however, almost half of the platform engineers are designing parts for the first wave only -- the lifecycle offset effect is clearly present. When making the same comparison to their supervisors, the trend improves, but not significantly. It should be expected to have the same supervisors yet several engineers, and the supervisor would be the coordination enabler between the engineers in the different projects.
- In terms of the Top Hat engineers, they were mostly unique to one wave with some sharing between waves, enabling different designs as expected. However, it would have been expected to have significantly more shared supervisors between both waves to enable coordination and learning.

Furthermore, another interesting finding was that in some cases common part numbers for the various projects were assigned to different engineers, especially if the projects belonged to another wave. A total of 49% of the shared parts between both waves presented this situation where the first engineer designed the part for the first project, and the second engineer was just copying the design for the following project (if suitable). However, it is unclear if the requirements for the second product / project were clearly understood by the first engineer, or even if the engineer was aware that the part was being shared with other projects. In the best scenario, both engineers should be closely working together to ensure the requirements for their respective projects were consistent in order to avoid avoidable divergence in any of the projects.

In addition to the low number of engineers designing parts for both waves, the key engineering coordinators for the main functional areas (Interior, Exterior, Chassis, Powertrain and Electrical) were also assigned only to their respective waves. Yet, their managers were responsible for both waves (maybe without equal attention), thus creating another possible functional engineering link between the two waves. The problem with this coordination mechanism is that the higher the coordination link exists in the organizational chart it is usually less effective because higher ranked personnel are often unaware of lower level design details. While the lifecycle offset between waves was a difficult situation to coordinate among different development projects, the team structure of the second wave also contributed to the lack of coordination as well. The engineering functional areas were relying on the individual design engineers to achieve common designs; however, achieving common design is the responsibility of the entire team.

The coordination of the integration between resources (waves) was enabled by the team organization as described in the organization chart (figure 3.5). The existence of a senior leader managing all the projects and the coordination mechanisms between both of the projects (similar processes, documents, dual reporting mechanisms) allowed for better coordination between the projects than between the different functional areas (i.e. for conflict resolution). However, engineering had the ultimate responsibility to develop the product and therefore assess if the part could be shared or not. Cusumano and Nobeoka (1998) also described these coordination mechanisms between projects (between project managers and by having a general manager above them) and between functional areas (by a functional manager or direct coordination between engineers), and according to their research the coordination between projects is usually stronger than the coordination between the functional areas.

Team organization can influence product commonality and divergence; however, it is very difficult to quantify the effect to commonality for a specific organizational scheme. Product commonality is another aspect in the well-known debate for functional or project oriented organizations. Overall product commonality accountability is usually assigned to the project manager; however, he/she is not responsible for designing the parts and ensuring they are functional and efficient for all projects. On the other hand, the individual design engineers are responsible for designing the parts, and the best mechanism to hold the engineer accountable for commonality is by assigning him/her to all of the affected projects, so that divergence will impact the engineer's own workload.

Other possible coordination mechanisms to improve the team organization concerns are the "dual responsibility system for engineers" implemented by Mitsubishi or the "differentiated matrix" (Cusumano and Nobeoka, 1998). In the dual responsibility system, the engineers are assigned not only to design specific components but also to take coordination tasks among projects. Each engineer is not only responsible for integrating a set of components into a vehicle, but also to coordinate the integration of the same component for different projects. In the differentiated matrix scheme, the functional areas are organized in such a way that some functional organizations develop standardized solutions for several products, while other areas can develop specific solutions to specific products. The product platform organization used in the project under study can be considered another variation of the differentiated matrix organization, just with a broader scale of a product platform or product family instead of individual products.

In summary, the widely used matrix organization in product development can be another factor that may influence the ability to achieve common designs. Common designs will ultimately be the responsibility of both the project manager and the design engineers, so the coordination mechanisms between project managers and between engineers must be in place to allow integration across projects and across different subsystems.

3.6.2.3. Economic Incentives Effect on Divergence and Product Commonality

The main reason for achieving common designs and developing product platforms and product families are the economic benefits, the most relevant being reduced engineering expenses, reduced investment in capital equipment, and the economies of scale driven by larger volume of products using the standardized solution. Using the same information from the sharing studies in section 3.5.3, a comparison between couples of derivatives or among a set of derivatives was performed to understand the product commonality effect of four different factors:

- Wave: What is the effect on product commonality for developing the products in the same "wave" of products? In other words, developing the products mostly in a concurrent manner. Besides the concurrent development, a common wave represents also a similar set of performance and customer requirements.
- Top Hat: Market effect of having the exact same exterior and interior styling execution compared to using the same product platform for products that have different exteriors.
- Region: Effect of the regional market requirements, i.e. preferred fuel economy compared to vehicle acceleration performance
- Plant: Effect of having the same manufacturing location and supply base on product commonality. Does it drive incremental commonality?

Table 3.15 below provides the comparison of these four factors and the corresponding effects on product commonality. Probably the most relevant economic factor to enable common parts was the decision to use common or different assembly plants. For a considerably large portion of the parts (or at least, the ones that represented the highest cost), different production locations for the supply base were required, meaning that additional investment would be required for each assembly plant and the corresponding supply chain, even if the designs were common. This situation may have enabled divergence since there was no clear economic reason to have a completely shared design if the supply base was going to be different.

Also, another observation from table 3.15 is that there are few situations where there is a common plant for the different derivatives; in general, common plants enable to increase the number of shared parts. In the case of the Egyptian Dog and Retriever, the total number of underbody shared parts increased significantly compared to other projects sharing common exterior styling. In the case of the Greek Dog, Greek Wolf, and Steppe Wolf the common plant even drove the largest number of shared components in a platform perspective, even if they have different styling and are being developed on a different timeline. Nevertheless, contradictory effects are found for the Mayan Dog and Wolf projects, where having a common assembly plant does not drive a significant improvement in commonality.

Derivatives		Factors				No of Common factors	Complete Vehicle		Underbody (Platform) Components only	
Lead	Follow on	Wave	Top Hat	Region	Plant		Shared Parts (Absolute)	Shared Parts (%)	Shared Parts (Absolute)	Shared Parts (%)
ED	GD	Common	Common	Different	Different	2	0.879	23.26%	0.346	19.69%
ED	MD	Common	Common	Different	Different	2	0.757	37.07%	0.230	27.18%
ED	R	Common	Different	Common	**Common**	3	0.542	**22.21%**	0.330	**35.17%**
ED / R	GW / SW	Different	Different	Different	Different	0	0.255	5.69%	0.137	8.46%
ED / R	EW / TW	Different	Different	**Common**	Different	1	0.278	7.16%	0.143	10.46%
ED / R	MW	Different	Different	Different	Different	0	0.206	5.58%	0.104	7.51%
R	GD	Common	Different	Different	Different	1	0.420	10.41%	0.284	16.55%
R	MW	Common	Different	Different	Different	1	0.309	13.51%	0.174	21.85%
GD	MD	Common	Common	Different	Different	2	0.906	26.91%	0.293	19.35%
GD	GW / SW	Different	Different	Common	**Common**	2	0.449	9.15%	**0.343**	**18.08%**
GD	EW / TW	Different	Different	Different	Different	0	0.195	4.23%	0.115	6.17%
GD	MW	Different	Different	Different	Different	0	0.235	5.48%	0.150	8.28%
MD	GW / SW	Different	Different	Different	Different	0	0.209	6.35%	0.135	12.62%
MD	EW / TW	Different	Different	Different	Different	0	0.169	6.11%	0.097	11.31%
MD	MW	Different	Different	Common	**Common**	2	0.174	6.95%	0.095	11.28%
GW / SW	EW / TW	Common	Common	Different	Different	2	0.536	15.40%	0.222	18.99%
GW / SW	MW	Common	Common	Different	Different	2	**1.000**	36.27%	0.322	30.84%
EW / TW	MW	Common	Common	Different	Different	2	0.632	24.69%	0.306	**37.95%**

Table 3.15: Effect of the different factors to the number of shared parts from a complete vehicle and a platform perspective. The absolute number of parts has been scaled to the maximum number. The percentage of shared parts represents the number of shared parts compared to the total number of parts between the two derivatives under consideration

To verify the significance of the different factors driving common designs, four design of experiments (DoE) were performed considering all of the above factors. These four DoEs correspond to the four measurements considered, two from a complete vehicle perspective and two from an underbody perspective, to eliminate the bias of all the common styling parts driven by the common styling execution. A summary of the results and the transfer equations are shown in table 3.16 and in figure 3.25. The DoE was prepared with the data from table 3.15. Each of the factors had two levels, Common or different, and these factors were translated into binaries: if a factor was common it was considered a "1" and if it was different it was considered a zero. The results of the DoE are summarized in table 3.16 and figure 3.25.

Measurement (Y)		Transfer Equation	R^2
Complete Vehicle	Shared Parts (Absolute)	Y= 0.215Wave + 0.354TopHat + 0.215	0.8521
	Shared Parts (%)	Y= 7.75Wave + 13.61TopHat + 5.75	0.774
Underbody (Platform) Components only	Shared Parts (Absolute)	Y= 0.104Wave +0.02Region+ 0.058TopHat - 0.048Plant + 0.122	0.8619
	Shared Parts (%)	Y= 15.22Wave + 9.375	0.6469

Table 3.16: Transfer equations for the four design of experiments.

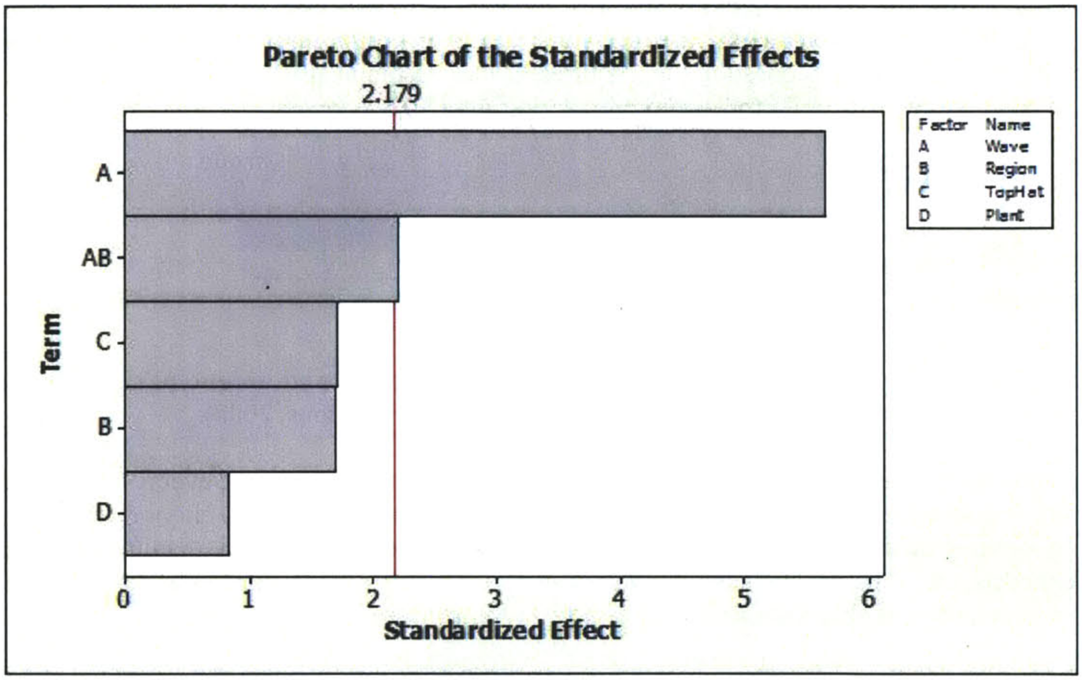


Figure 3.25: Significance of each of the different factors (enablers) on the commonality of the vehicle and the platform.

Some conclusions can be derived from the data and the analysis above:

- From an overall vehicle perspective, the most significant factor is the common Top Hat or styling (interior and exterior). The other significant factor is the common schedule, meaning that they are being developed as part of the same wave.
- After eliminating the upperbody or Top Hat components, the common timeline is the only significant factor in commonizing.
- The effect of having common plants (or region) is an enabler for improving commonality too; however, it is not statistically significant in this case study. Even so, having a common manufacturing location was perhaps the factor that actually drove commonization and convergence behavior. These factors had also an important economic impact, different from the rest of the common parts whose advantage was having common development.

In summary, given that the most important factors that drove a larger number of common parts were the wave and the Top Hat, it can be concluded that customer requirements took a more relevant role than actual manufacturing constraints, which usually lever lower costs. Achieving common designs allowed for reduced engineering expense. Nevertheless, the economies of scale and the reduced capital investment was constrained only to inexpensive parts that could be cheaply produced in one location and easily and cheaply transported to serve all the different assembly plants. The effect of having multiple manufacturing locations resulted in more efficient designs for each region, because capital investment in two or three locations was planned regardless if the designs were common or not, and therefore enabled product divergence.

3.6.2.4. Cousin Parts: Partial Commonality as an Economic Incentive

The design engineers have the task to create efficient solutions for a wide range of products and requirements. Most of the commonality indexes compare only two extreme perspectives: common or unique parts. However, there is a gray area for cousin parts or similar parts (Boas, 2008) and the degree of commonality is also a valid question that requires further decomposition for answers. This scheme of "degree of commonality or similarity" is represented in figure 3.26.

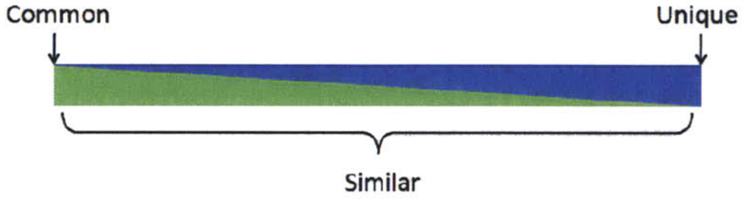


Figure 3.26: Cousin or similar parts definition. Between common and unique parts, there can be a wide range of parts with partial commonality. (Reproduced from Boas, 2008).

Design commonality measurements depend significantly on the method used. Several indices have been proposed to measure the commonality index (Thevenot & Simpson, 2006, 2007) and the result is different depending on the process and inputs used: common components, common manufacturing process, component costs, etc. The present case study presented two different indexes to assess commonality, and resulted in very different observations.

One of the main characteristics of the corporate commonality index (C,M,N,U) is the definition of a modified part. These "modified" parts are an example of a cousin part, as they are parts that share some common components; however, the complete assembly differs. Figure 3.27 graphically illustrates the definition of a cousin part. The highest level of assembly of the part can have different variations driven by different components or modules; however, there can be significant reuse of sub components in lower levels of assembly. In the case study, many of the parts may be considered cousin parts, either the ones that are modified (M) or unique (U), however, determining if they are "cousin" parts is context dependent.

Cousin parts are relevant for economic reasons. Even if the part can not be entirely reused, a significant portion of the design and the manufacturing process might be reused, and designing a new part would not represent a completely new development. In many situations in the case study, it was a better trade-off to create a new part number (and therefore, have divergence) in order to have a more efficient design to the respective set of requirements because the development of a new part would not represent a completely new development or a significant incremental capital investment. Nevertheless, given the commonality index definition, a new cousin part will always represent divergence, yet without representing a significant economic loss.

The scheme above also illustrates the relevance of product and component modularity, which is closely related to product commonality (Fixson, 2007). The need for increased product variety in mass produced products (like the automotive industry) has lead to a new framework known as mass customization (Pine, 1993) that uses product modularity to achieve a larger variety of products in a cost effective way. Figure 3.27 shows graphically the modularity concept. By adding, deleting, or modifying specific components, a new part (or product) can be assembled to satisfy a different set of requirements with minimal incremental development and investment.

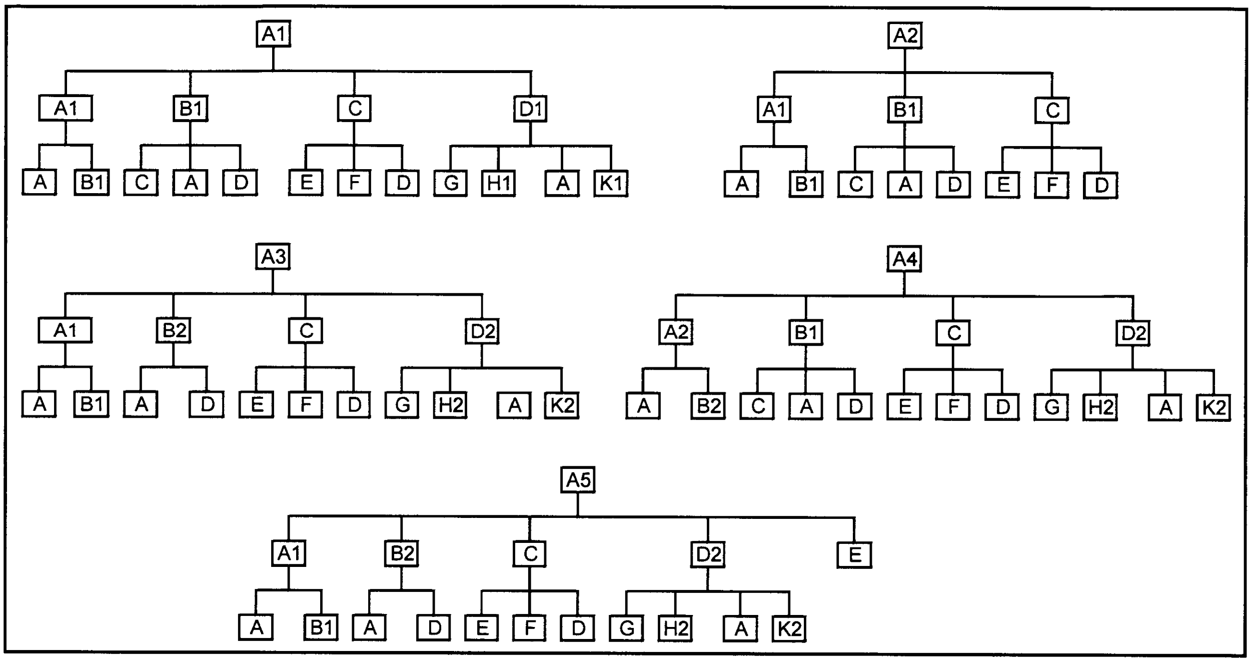


Figure 3.27: Components tree for five different cousin parts labeled A1 to A5 (highest level). All the parts (level 1) are further decomposed in two assembly levels, showing several shared modules (level 2) or components (level 3)

The main philosophy of the product platform emphasized the use of a range of different parts to achieve the different requirements for each of their derivatives. The design philosophy stressed the following principles:

- Principle 1: Common vehicle architecture (unibody), common bill of design, common bill of process and common underbody locators.
- Principle 2: Common system architectures: suspension, underbody and powerpacks.
- Principle 3: Different parts within common architectures to support the different vehicle requirements: reinforcements, gages, tuning, materials, etc.

To economically create varied parts to support different vehicles, the concept of product modularity and cousin parts was used as another tool to achieve economies of scale. Different part variations were created to support a wide range of vehicle features, such as different engines and transmissions, distinct exterior and interior styling or specific drive configurations; these parts usually had common and unique modules or components as explained in figure 3.27.

Since most of the economies of scale could be achieved with a common supply base and assembly locations, designing common parts was not as relevant as achieving economic efficiencies. This can be inferred from the high number of unique parts found in the case study. However, as parts become more complex and have more components, the opportunity for having common parts between derivatives and variants is less likely. Figure 3.28 shows the distribution of the parts' cost for the shared and unique parts, demonstrating that most of the shared parts were usually cheap and relatively simple; the unique parts had a wider distribution and represented more complex assemblies designed for specific applications.

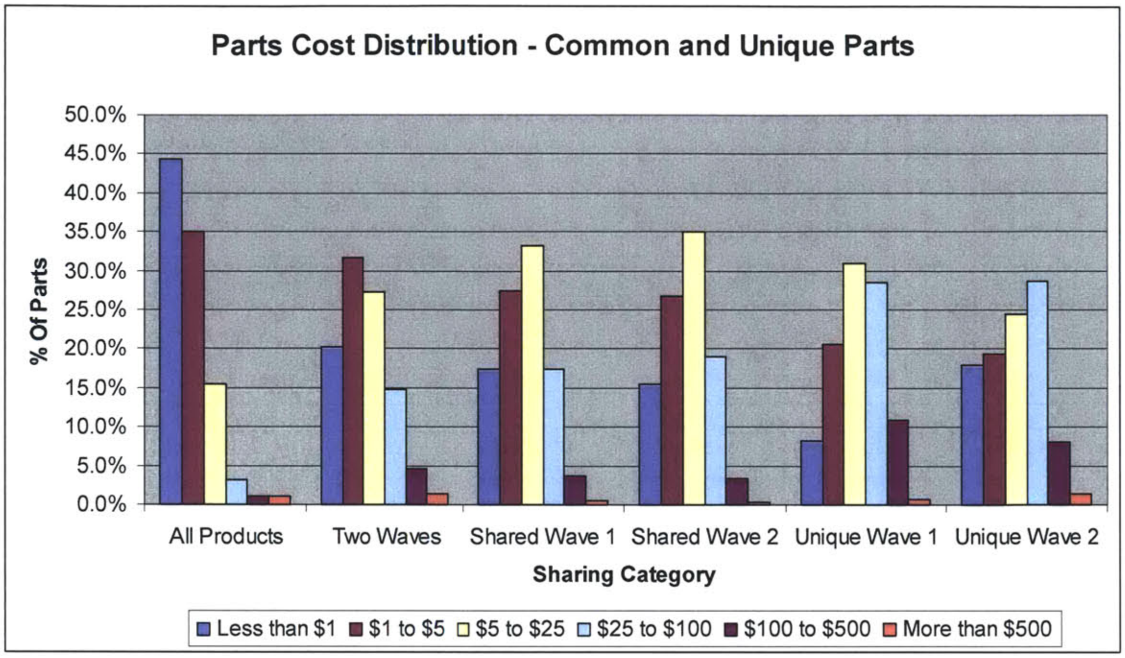


Figure 3.28. Distribution of the costs for the shared and unique parts for both waves.

The design strategy above facilitated the creation of tailored solutions for each of the different build combinations, and therefore caused a relatively high number of part variations in the assembly plants. It could be argued that the additional part variety could impact manufacturing performance or quality. However, MacDuffie, Sethuraman and Fisher (1993) studied the effect of part variety to manufacturing performance and found that they do not have a negative impact on these dimensions. Furthermore, Cusumano and Nobeoka (1998) listed the following strategies to minimize the impact of parts variety in manufacturing:

- Flexible manufacturing
- Low inventory production techniques
- Parallel assembly or production lines
- Computerized scheduling and planning systems
- Flexible automation
- Fast set-up equipment.

Their conclusion was that there are more solutions to minimize the impact of variety in manufacturing than solutions in design (like modularization) to minimize the impact of creating more parts to fulfill the range of product requirements. Having this wide range of alternative solutions for handling incremental variety may have also enabled divergence.

Summarizing, the product platform philosophy, the partial economic incentives for creating cousin parts, and the wide range of manufacturing solutions to handle variety contributed to divergence. It is impossible to quantify the economic effect of the cousin parts compared to common parts, however, by analyzing the distribution of the common and unique parts costs it can be inferred that the end goal of the engineers was designing efficient parts for their application instead of designing common solutions that may drive higher costs or increased non required functionality. Also, understanding that most of these parts were sharing components in lower levels of assembly, the actual product commonality could be higher if it were computed from the final component leaves instead of a predefined level of assembly within the branches of the bill of materials.

3.6.3. Change Propagation and Divergence

The focus of the present work is to understand key actions that a project manager could implement to minimize divergence in platform product development, so most of the discussion has been centered in the process and the enablers for divergence; nevertheless, the nature of the parts and the product being designed can be another relevant factor that can influence product divergence. In a similar analogy, while the product development process can enable divergence naturally, the nature of the product can enable "natural" divergence too.

Eckert et al (2004) developed a framework for change propagation based on the nature of the product. They first differentiate between two types of changes: the "emergent" changes, caused by the state of the design and the "initiated" changes that are started by an outside source like a customer or the manufacturer. A similar classification was proposed in figure 3.22 to classify the sources of divergence, where the changing requirements, new technologies and component obsolescence could be classified as "initiated" changes, while the learning in product development and operations can be related to the emergent changes. In this classification, the emergent changes have an inherent negative connotation.

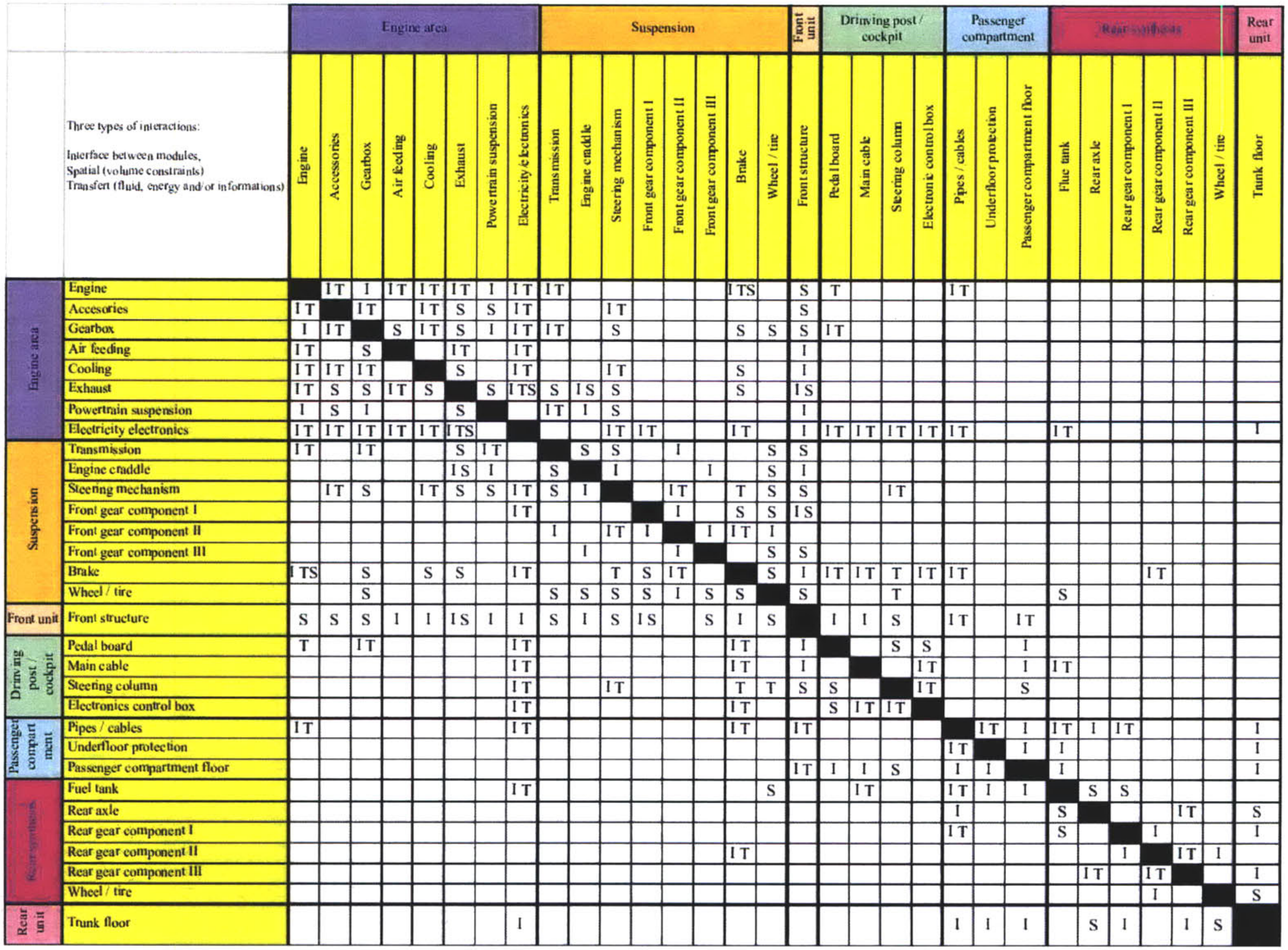
Change propagation is a characteristic of complex products that are usually highly interconnected, where a change in one of the parts can create a change in other parts in order to satisfy the product requirements. The degree of interconnection within the subsystems or parts of a product is dictated by its product architecture: the scheme by which a function of a product is allocated to its physical components. Ulrich (1995) acknowledged the deep impact of product architecture has on product change, product variety, component standardization and product development management. Consequently, understanding the product architecture can give further insights in the phenomenon of change propagation.

Product architecture can be categorized in two opposite schemes: integral or modular architecture (Ulrich, Eppinger, 2004). The modular architecture has usually a 1:1 mapping of functions to physical elements and the integral architecture usually has a complex mapping from functional elements to physical elements (Ulrich, 1995). The automotive industry also has a specific product architecture differentiation and two main architectures are acknowledged: body on frame or body frame integral construction; and these represent the relationship between the chassis (the platform) and the body (the Top Hat) of the vehicle.

A useful tool to represent the product architecture is the design structure matrix (DSM) that reproduces in a square matrix the relationship between all the different subsystems (Eppinger et. al 1994). Figure 3.29 reproduces Mahmoud-Jouini and Lenfle's (2010) proposed DSM for the platform of a vehicle, in a similar definition of a platform as in the present case study. The matrix details three types of interdependencies between systems:

- Interface (I) – The components are attached together
- Space (S) – Interaction between components due to the limited space
- Transfer (T) – Transfer of information, energy or flow between components

The conclusion of analyzing the DSM of a vehicle platform is the large amount of interactions between components, especially in the engine area, showing a mostly integral architecture. For modular architectures, the DSM has more blank cells, meaning less interactions between components and interactions tend to cluster in groups. Choosing between one and the other type of architecture has several consequences for the firm (Ulrich, 1995), nonetheless, in terms of components standardization or ease of change, the modular architecture behaves in a better way as it facilitates both characteristics. Furthermore, the low number of shared parts is another sign of the integral architecture of the product that has to be managed accordingly. The degree of modularity may be limited by the amount of functional performance required as modularity generally introduces some amount of inefficieny (Holtta and de Weck 2007)



	Three types of interactions: Interface between modules, Spatial (volume constraints) Transfert (fluid, energy and/or informations)	Engine area								Suspension								Front unit	Driving post / cockpit				Passenger compartment			[illegible]						Rear unit
		Engine	Accessories	Gearbox	Air feeding	Cooling	Exhaust	Powertrain suspension	Electricity électronics	Transmission	Engine craddle	Steering mechanism	Front gear component I	Front gear component II	Front gear component III	Brake	Wheel / tire	Front structure	Pedal board	Main cable	Steering column	Electronic control box	Pipes / cables	Underfloor protection	Passenger compartment floor	Flue tank	Rear axle	Rear gear component I	Rear gear component II	Rear gear component III	Wheel / tire	Trunk floor
Engine area	Engine	■	IT	I	IT	IT	IT	I	IT	IT						ITS		S	T				IT									
	Accesories	IT	■	IT		IT	S	S	IT			IT						S														
	Gearbox	I	IT	■	S	IT	S	I	IT	IT		S				S	S	S	IT													
	Air feeding	IT		S	■		IT		IT									I														
	Cooling	IT	IT	IT		■	S		IT			IT				S		I														
	Exhaust	IT	S	S	IT	S	■	S	ITS	S	IS	S				S		IS														
	Powertrain suspension	I	S	I			S	■		IT	I	S						I														
	Electricity electronics	IT	IT	IT	IT	IT	ITS		■			IT	IT			IT		I	IT	IT	IT	IT	IT			IT						I
Suspension	Transmission	IT		IT			S	IT		■	S	S		I			S	S														
	Engine craddle						IS	I		S	■	I			I		S	I														
	Steering mechanism		IT	S		IT	S	S	IT	S	I	■		IT		T	S	S			IT											
	Front gear component I								IT				■	I		S	S	IS														
	Front gear component II									I		IT	I	■	I	IT	I															
	Front gear component III										I			I	■		S	S														
	Brake	ITS		S		S	S		IT			T	S	IT		■	S	I	IT	IT	T	IT	IT						IT			
	Wheel / tire			S						S	S	S	S	I	S	S	■	S			T					S						
Front unit	Front structure	S	S	S	I	I	IS	I	I	S	I	S	IS		S	I	S	■	I	I	S		IT		IT							
Driving post / cockpit	Pedal board	T		IT					IT							IT		I	■		S	S			I							
	Main cable								IT							IT		I		■		IT			I	IT						
	Steering column								IT			IT				T	T	S	S		■	IT			S							
	Electronics control box								IT							IT			S	IT	IT	■										
Passenger compart ment	Pipes / cables	IT							IT							IT		IT					■	IT	I	IT	I	IT				I
	Underfloor protection																						IT	■	I	I						I
	Passenger compartment floor																	IT	I	I	S		I	I	■	I						I
[illegible]	Fuel tank								IT								S			IT			IT	I	I	■	S	S				
	Rear axle																						I			S	■			IT		S
	Rear gear component I																						IT			S		■	I			I
	Rear gear component II															IT												I	■	IT	I	
	Rear gear component III																										IT		IT	■		I
	Wheel / tire																												I		■	S
Rear unit	Trunk floor								I														I	I	I		S	I		I	S	■

Figure 3.29: Platform DSM (Adapted from Lenfle et. al (2010)

What the platform DSM provides can also complement the understanding of the effects of product change, specifically related to divergence. The case study has emphasized the enormous effect that the engine can have into their surrounding parts; this is further shown in the DSM given the highly coupled design of the adjacent components. A clear example was the relatively small change in the position of the engine that drove significant change and divergence in the second wave of derivatives. The integral architecture enabled change propagation into other systems and therefore, enabled divergence as well, however this divergence is difficult to avoid since it is a characteristic of the product itself.

The change propagation framework developed by Eckert et Al. (2004) characterizes the different components into the following categories based on how the change is propagated:

- Constants: components that are unaffected by changes
- Absorbers: components that can absorb more changes than the ones they generate
- Carriers: components that absorb and generate changes with a similar frequency
- Multipliers: components that generate more changes than the ones they absorb

It is intuitive to the organization when the nature of automotive components is categorized in the framework above. Some components are more likely to change their design than others and what are the usually affected parts when these change. For example, the engine is a clear example of a multiplier, so the project manager will always try to avoid changes to the engine to evade change propagation. On the other side, the wiring harnesses are a clear example of an absorber, as they are usually affected by relatively small changes, however, they are easy to fix.

The relevance of product architecture, change propagation and divergence into the present case study is the proposed framework of dividing the product into two major systems: the platform and the Top Hat system. Based on the frequency of decisions and discussions needing both project managers agreement on the changes, the conclusion is that this split is possible and beneficial. Furthermore, Muffatto (1999) argues that there is only a 10% overlap between the upperbody and the underbody design. Using a complete vehicle DSM (instead of the platform or underbody as the one copied here) can give further insights into the key components that can propagate changes from the upperbody to the underbody and vice versa.

From the relative amount of design churn of the components (how often and how many changes) in the change control, and the number of changes to the underbody driven by the upperbody design, some of the underbody components functioned as absorbers for the changes. The underbody components that can be categorized as the "platform absorbers" for upperbody changes were the cross car beam (Instrument panel structure), the wiring harnesses and the radiator support. These parts are difficult to characterize as either platform or tophat components as they have characteristics of both: the team acknowledges their design will not be complete until the upperbody is designed and the styling changes usually propagate to them; however, they are not visible to the customer and can potentially be shared (or some portions of them). These parts should have a separate category as "transition parts" and should be managed in a special way. Future work could quantify the various components and subsystems of both the Top hat and the underbody platform on the absorber to multiplier spectrum using the CPI (change propagation index) developed by Giffin et al. (2009). This would require a detailed analysis of engineering change request records which is beyond the scope of the present thesis.

3.7. Case Study Summary

The case study described the progress of a complex automotive product family over three product development phases with an emphasized focus on product commonality behavior over time. A significant portion of the available literature in product platform development had described the relevance of product family planning and the key advantages of commonization; however, it has been centered on the product planning phase and the case study presented some of the key issues during more advanced product development phases, specifically related to divergence, i.e. the loss of design commonality over time.

The automotive product family was developed through the evolution of an existing product platform whose scope was widened to have a global scale, and represented the convergence of five legacy product platforms (underbodies) into one. This convergence of several product lines into a single platform, the increased scope from a regional to a global perspective, the fulfillment of more stringent requirements and the development of completely new exteriors and interiors for all the vehicles represented a very aggressive development plan. In order to spread the risks of product development, ensure a continuous stream of product launches and reduce the calendarized workload and expenses, the product family was executed by staggering the projects and grouping them by market segments into product "waves".

To overcome the issues related with separating and staggering the projects into these waves and enhance coordination between waves, a new organizational structure was utilized. Its key feature was separating the vehicle projects into two separate projects for the underbody and the upperbody of the vehicle. The new structure aimed to emphasize the commonization of the underbody or platform components while allowing differentiation of the upperbody or Top Hat components by clearly dividing those workstreams into two (or more) different projects with their respective project manager. Furthermore, these different workstreams were already part of the product development system which reinforced the need for the new organization. Inside the company, the names of the projects and products were intuitive and it was relatively easy to identify all the vehicles that were part of the product family and created a perception that these vehicles shared significant features of their design.

The proposed structure proved to be beneficial for the design and development engineers that were organized according to the existing structure based on the vehicle subsystems; however, the integration activities like manufacturing, project management or vehicle engineering were deeply impacted by the new organization. The integration activities were now required to differentiate if the task or change was related to the underbody or the upperbody (based on the affected parts), creating a virtual triple report relationship: one to the tophat project, other to the platform project and finally to their respective functional department. Besides the more complex reporting structure, the actual integration of the projects now required an additional step, creating additional workload for them. The main benefit of the structure was the use of an unchanged product development system and the existing matrix organizational structure, creating a mostly favorable structure overall with some burden on the integrative activities.

Product commonality was expected to be a key improvement area with the organizational structure. Throughout the project development, divergence was present in the design, and the overall product commonality decreased throughout the project as a consistent behavior for the underbody and the upperbody system. Design commonality was found to be highly dependent on the assessment method and proved to be information intensive given the high number of different derivatives in the product family. Regardless of the assessment method, the number of unique parts increased over time, and the number of shared parts decreased. The existence of the platform project enabled some commonization opportunities, nevertheless, divergence always dominated convergence throughout the project.

The divergence behavior remained virtually constant throughout the project, and it was present in all the assessed product development phases and projects. Divergence can be minimized if the appropriate amount of attention is given to all the projects concurrently. The lead derivatives and the "control model" parts usually received increased attention from the project managers, and the increased attention gradually shifted from the high cost parts to all the parts in line with the increasingly available workforce and the design progression.

Besides the actual commonality data, the case study further described some of the actual design changes that drove divergence, and some of the probable causes. The framework of divergence sources and enablers developed by Boas (2008) was complemented. Divergence will be present based on the normal design progression dictated by the product development process and also based on the product architecture and change propagation. These two factors can not be easily modified based on the complexity of the product and the development process. The other sources of divergence: changing requirements, new technologies and component obsolescence can be controlled by the project manager and divergence should be a result of all the different design decisions and tradeoffs made by the project manager.

The project manager can further influence the divergence behavior by improving the coordination between projects and within the project. A relevant lack of coordination which impacted divergence significantly was the project hand off from the planning to the executing team which also represented the use of the corporate tools and actual engineering estimates instead of surrogate data. Commonality is always the highest at the beginning of the project where common parts are an easy assumption to make. Overcoming these realities would require the executing team to be involved earlier in the project, however, that would imply higher engineering expenses and should be carefully assessed. The lack of coordination can be influenced by the processes (project accounting or a limited project assessment scope for the control models) or the team itself (few engineers working on all the projects, unclear assignment to either the Top Hat or the platform, very limited integration coordination between waves).

The other relevant area where the project manager can influence divergence behavior is carefully assessing the economic incentives that could influence commonality. Divergence is often understood as a negative situation; however, divergence can be beneficial to the project if it can drive lower project costs. Four different factors were investigated as probable contributors to product commonality: shared engineering (common exterior styling or same vehicle wave) and shared manufacturing (same region or same assembly plant). The number of shared parts was found to be largely influenced by the common vehicle segment and common design timeline (same wave) compared to the opportunity to share the assembly location and the supply base. Yet, the economic incentives can be also partially achieved by the use of cousin or similar parts, which have proven to be a more effective solution as more solutions have been readily available to manage the wide range of requirements at the manufacturing end (with scheduling systems, flexible manufacturing, etc) rather than the design end (by creating modular designs or common solutions).

The high number of derivatives and projects in the canine product family enabled analogies and comparisons between different projects that were developed in a homogeneous environment with equivalent processes, schedule and management pressure. While these observations can not be generalized to be the expected behavior in other projects, several features of divergence behavior were found in all the projects of the canine product family. The two key contributions of the case study are the deeper understanding of divergence over time and within different PDP phases, and the relative influence of the divergence enablers to the overall product commonality.

The platform project manager must actively manage product commonality based on the available information. Product commonality is not an objective but a way to achieve more efficient product development. To achieve an efficient project, conscious and timely decisions must be lead by the project manager, which can range from relatively simple engineering changes to the addition or deletion of an entire product from the product family. Achieving a balanced "iron triangle" for all the different projects requires equal attention to all the projects under their scope, however, the time component in the iron triangle is usually the component that influences the behavior of the project the most, either in the form of lifecycle offsets or by the unconscious behavior of focusing too much on the lead project based on its urgency and the increased available time for its derivative products.

4. System Dynamics Model Development and Calibration

4.1 Model Structure and Objective

As concluded in the bibliographic research, system dynamics has emerged as a valuable tool for project management. Most of the existing project management tools do not capture the dynamic nature of projects, as they have to adapt as the project progresses in order to achieve the project's objectives. In addition, the bibliographic research also showed that platform product development has been addressed mostly as a static phenomenon, where the reuse benefits implicitly assume the common designs can be achieved; however, divergence is a common phenomenon in the development of product families.

While most of the product platforms literature has addressed the tradeoff between product commonality and product costs and product performance, the effect of designing common components on the project schedule has been only tangentially addressed by acknowledging that common components usually require more time to develop. Therefore, the main objective of the system dynamics model is to develop a conceptual framework to understand the effects of divergence on the project schedule of a set of product development projects derived from a common platform.

The model will include the simulation of two product development projects being designed concurrently, where each project will be the sum of the unique and common parts. The model was developed using the Vensim PLE ® software (Vensim ® DSS for Windows Version 5.10c, Ventana Systems, Inc) designed to simulate and analyze system dynamic models by constructing their causal loop and stock and flow diagrams.

The following sections will further develop the project features and the proposed stock and flow structure for the model that will be calibrated and correlated to the case study explained in section 3. The final model will include product commonality, divergence and lifecycle offsets to fully understand the tradeoff between product commonality and project schedule, and the effect to the individual projects in the product family to later explore possible management alternatives to achieve a better project schedule.

4.2 Description of the Basic Rework Cycle

The heart of the system dynamics models for project management is the rework cycle. A representative example of this cycle is reproduced in the diagram below (Lyneis, Cooper, Els 2001). As mentioned in the literature review section, the basic rework cycle theory is based on the fact that project tasks may look as though they are completed, nevertheless, they may not be truly completed at all since some errors may be found in downstream activities in the project and they may have to be done again, causing rework. The “work to be done” stock is influenced by three main factors: the amount of people, their productivity and the "quality" of the work being done. "Quality" is the fraction of tasks done correctly; the tasks not done correctly flow to the undiscovered rework and are later discovered and added to the work to be done, closing the rework cycle.

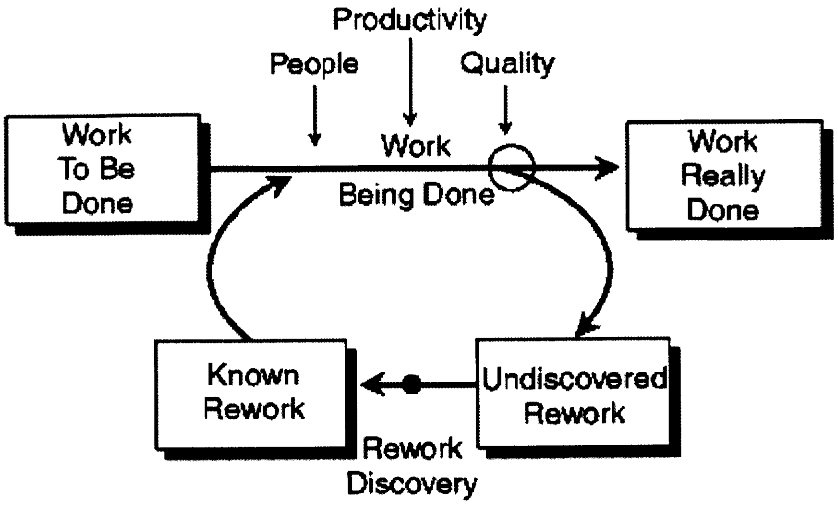


Figure 4.1 – Basic rework cycle (Lyneis, Cooper, Els, 2001)

A similar rework cycle will be used in the simulation model to understand the dynamics generated by commonality divergence in a multiple project management environment, meaning that similar stocks of work will be used for each project to describe the project dynamics. Additional features will be added to the model to better correlate to the case study.

4.3 Differentiation between Design and Integration Activities

The first model feature is the differentiation between two types of work: the work related to the development of the vehicle components, performed by design engineers (called "design work"), from the work related to the integration of all these parts into the vehicle (called "integration work"). The nature of the work performed by these two activities is different and therefore correlate differently to product commonality.

On one hand the individual design engineers have to balance or tradeoff all the different attributes for each of their parts: cost (variable cost, development cost, tooling cost), weight, functionality (proper function, part features, performance, manufacturing process feasibility, etc) and quality (customer satisfaction, craftsmanship, robustness, etc) and are concerned by the different design variants of their own components. On the other hand, the integration activities usually have to balance one aspect or attribute for the overall product (like weight, variable cost, tooling costs, NVH – Noise Vibration and Harshness, product safety, ride quality, towing capability, overall appearance, etc) and they require to understand how each of the parts contribute to the overall product performance, this means that their work is more related to the total number of product combinations which has a geometric / exponential behavior as seen in the introduction (figure 1.6).

While the nature of the design and the integration work is different, they are dependent on each other. Once the project starts, a complete set of product requirements is cascaded to the design activities in order to start the conceptual design of the different components which are later compiled and validated by the integration activities. The work usually flows from the design activities to the integration activities, which frequently have to return the work to the design activities if it was not done correctly, acting as a verifying entity for the prior design work. An additional relationship between these activities is the need for all (or most) of the design work to be completed to an agreed level of delivery (for example a point in time design freeze for a project review) in order to progress adequately the integration workstream.

The sequential dependency of the design and integration work was modeled with the structure detailed in figure 4.2 which represents the two different stocks of work (or tasks): the design work and the integration work. Both of these stocks are consumed by its corresponding work rates that are proportional to their own resources and productivity. The integration work can not be completed until the design work is completed, so the integration work rate depends on the completion of the design work. The dependency on both workstreams was modeled as a function of the progress on the design work by multiplying the integration work rate by the fraction completed by the design work. Only by completing 100% of the prior design work the model will allow the integration resources to achieve their full potential work rate.

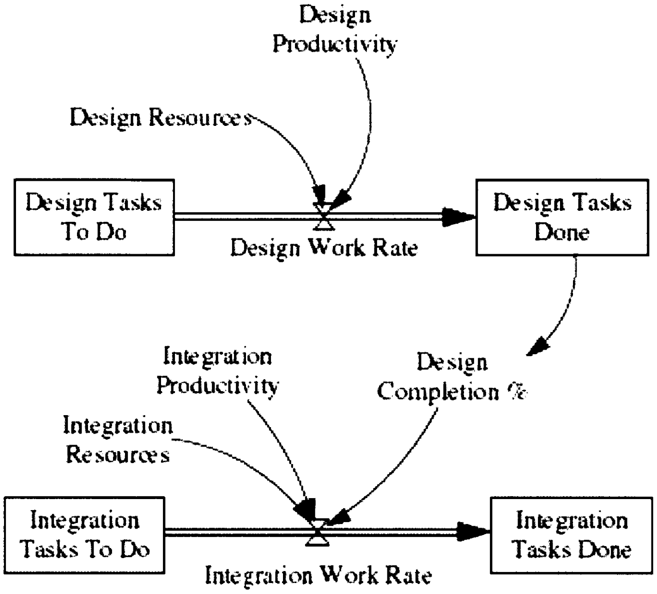
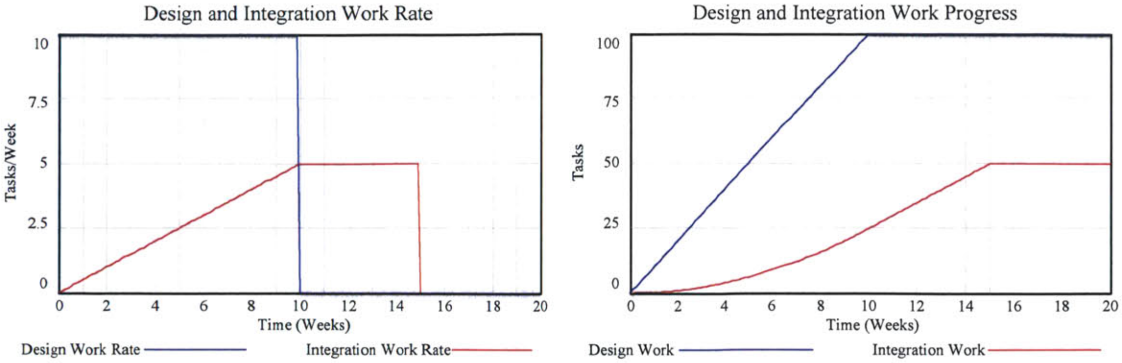


Figure 4.2: Design and Integration workstreams and their relationship.

A common example of this relationship is the construction of engineering prototypes. The assembly of a prototype can only be progressed proportionally to the parts that are available, and the construction will be completed only when all the parts are available. In other more strict scenarios, all the prior activities must be completed before the integration workstream can even start their own tasks.

A simple simulation was settled up to understand a priori the behavior of this sequential or dependent relationship. The hypothetical project requires 100 design tasks and 50 integration tasks, and has a staff of 10 design engineers and 5 integration engineers. If each person is able to perform 1 task per week (productivity), we would expect the project to last for 10 weeks if the tasks could be done in parallel. However, the integration activities are dependant to the design activities which must perform some work first, so their progress can not be higher than the progress of the design activities. The worst case scenario would be if all the design tasks and integration tasks were completely sequential and the duration would then be 20 weeks: 10 weeks to complete the design tasks and 10 weeks to perform the integration tasks. Figures 4.3 and 4.4 show the results of the base simulation that required a total of 15 weeks to complete the project, 50% more time than the expected logical time if the design and integration tasks could be performed in parallel.



Figures 4.3 and 4.4: Design and Integration workstreams: simulation results. Figure 4.3 shows the progress of the total tasks and figure 4.4 shows the rate of work completion.

On the one hand, based on this simple simulation, the sequential nature of the tasks makes the project impossible to finish on the expected time if the tasks could be performed in parallel. On the other hand, if the tasks are sequenced and performed by a different workforce, there will always be unused workforce capacity: early for the integration activities and late for the design activities. To optimize capacity, some of the options that project managers utilize in actual development projects are:

- Increase the work rate: either with increased workforce or with higher intensity (with overtime or with increased productivity). Usually the integration tasks are more prone to schedule pressure given the closer deadlines.
- Avoid the unused capacity by ramping up the integration resources later in the project or by ramping down the design resources earlier. Also, if the workforce is capable of doing so, they could perform both types of activities and support both workstreams when critical.
- Agree to an acceptable completion level and/or earlier deadline for the design resources. An example is freezing the designs to an agreed level throughout the design stages (for example a conceptual design or a system level design).

By using the strategies above, the workforce capacity can be optimized and the completion offset between the design and the integration activities can be minimized. Using this simple system dynamic model, additional runs were performed to quantify the relative improvement of using the strategies above, the summarized results are reproduced in table 4.1, and some of the simulation graphs are reproduced in figures 4.5 and 4.6.

This simple setup allows making some rough managerial recommendations on actions that could improve the project performance and provides a good example of the tension of the project "iron triangle". First, for reducing overall project time, increasing the work rate in the integration resources is more effective than increasing the work rate for the design resources. Defining if additional workforce or working overtime is more effective would depend on the cost of both alternatives. Second, to reduce workload and reduce project time, a lower completion level (80%, following the Pareto rule) is a common method to complete the project earlier, although, the risk of doing so has to be understood and managed by the project stakeholders. Finally, the strategy of adding the resources when needed is also an effective strategy to optimize workload, but the project would require more time to be completed.

	Design			Integration							
	Total Work [Tasks]	Resources [People]	Productivity [Tasks/People*Week]	Total Work [Tasks]	Resources [People]	Productivity [Tasks/People*Week]	Design Time [Weeks]	Total Time [Weeks]	Design Workload [People*Weeks]	Integration Workload [People*Weeks]	Total Workload [People*Weeks]
Base Scenario	**100**	**10**	**1**	**50**	**5**	**1**	**10**	**15**	**150**	**75**	**225**
Increase design resources 50%	100	15	1	50	5	1	6.67	13.33	200	66.67	**266.67**
Increase integration resources 50%	100	10	1	50	7.5	1	10	**11.67**	116.67	87.5	**204.17**
Increase design productivity 20% (i.e. work on Saturday or extra hours)	100	10	1.2	50	5	1	8.33	14.17	141.67	70.83	212.5
Increase integration productivity 20% (i.e. work on Saturday or extra hours)	100	10	1	50	5	1.2	10	13.33	133.33	66.67	200
Allocate design resources only for 10 weeks	100	10	1	50	5	1	10	15	100	75	175
Start integration work on week 5	100	10	1	50	5	1	10	17.5	175	62.5	237.5
Combine two actions above	100	10	1	50	5	1	10	17.5	**100**	**62.5**	**162.5**
Agree on 80% completion as an acceptable level	100	10	1	50	5	1	10	**14**	140	70	210
80% acceptable level and increase design resources 50%	100	15	1	50	5	1	6.67	12.67	190	63.33	253.33
80% acceptable level and increase integration resources 50%	100	10	1	50	7.5	1	10.00	**10.67**	106.67	80.00	**186.67**
80% acceptable level and increase design productivity 20%	100	10	1.2	50	5	1	8.33	13.33	133.33	66.67	200.00
80% acceptable level and increase integration productivity 20%	100	10	1	50	5	1.2	10.00	12.33	123.33	61.67	185.00
80% acceptable level and allocate resources when needed	100	10	1	50	5	1	10.00	**16**	100	55.00	**155**
80% acceptable level and allocate resources when needed with 50% integ product increase	100	10	1	50	7.5	1	10.00	13.87	100	66.53	166.53

Table 4.1: Multiple scenarios of managerial actions to improve resources use and project time.

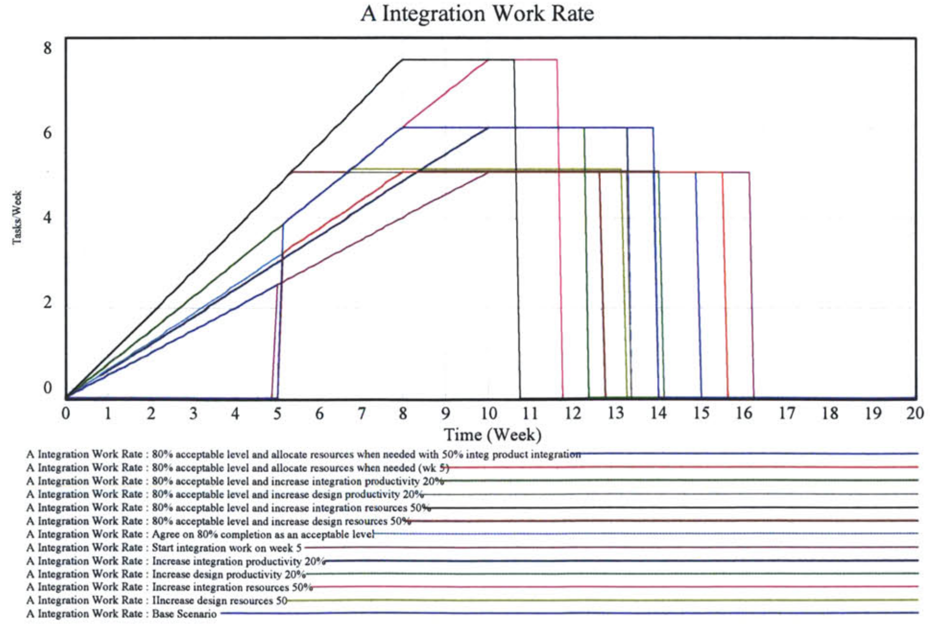


Figure 4.5: Simple simulation results: Integration Work Rate

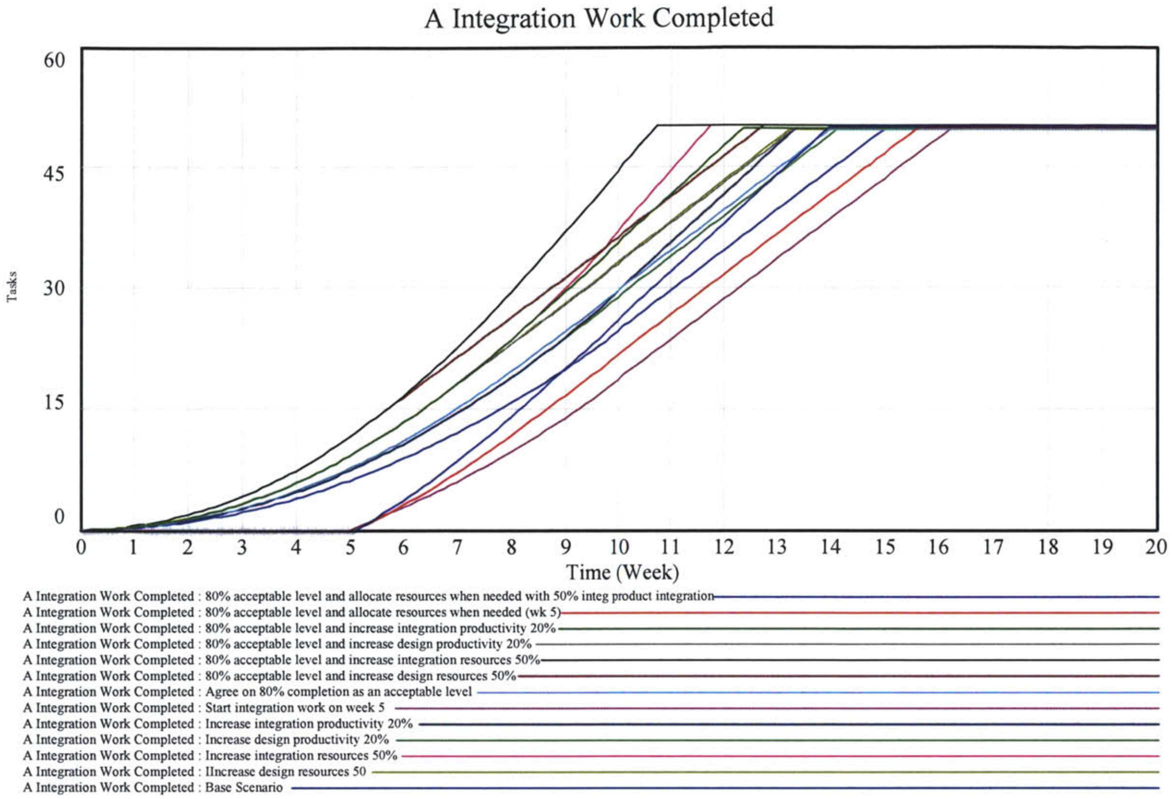


Figure 4.6: Simple simulation results: Integration work completed (for a total of 50 tasks)

To better understand the nature of the tradeoff, the graphs in figure 4.7 represent the simplified behavior of the project over time for the dependent design and integration tasks. The behavior of the work and the work rate are correlated as the work rate represents the speed the work is done. The work to be completed is the integral of the work rate over time, this is represented in figure 4.7 where a constant work rate yields a linear progress and a linear rate represents a quadratic progress (also seen in figure 4.6). Using these figures, the behavior of the work completed and the work rate will be derived analytically

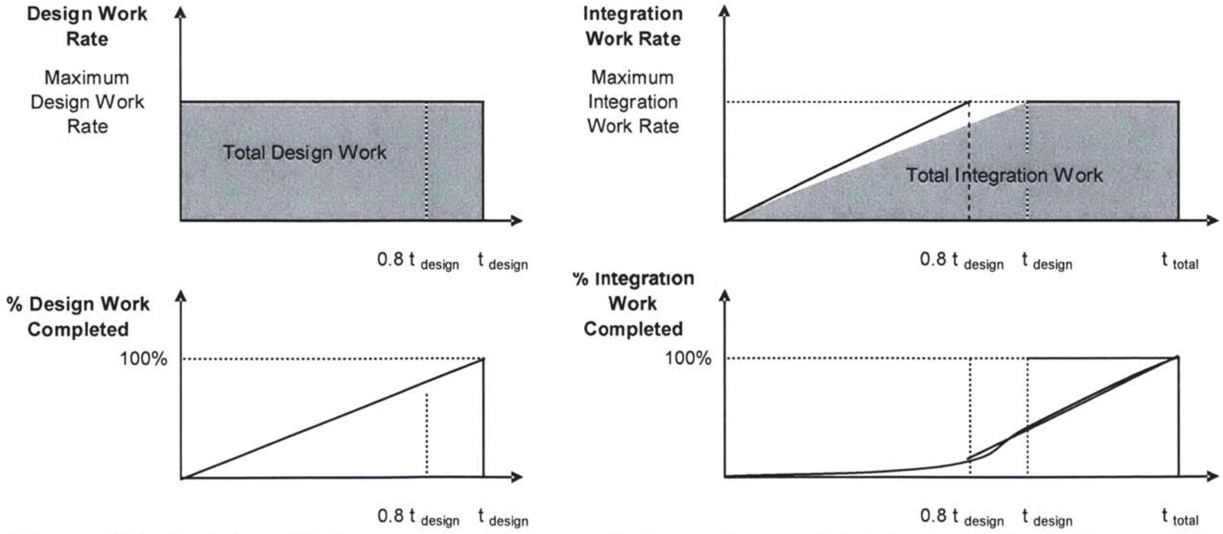


Figure 4.7: Relationship between work rate and the work completed for both the design and the integration work.

Given:

DW – Design Work
IW – Integration Work
t_{design} – Time required to complete the design work
DR – Design Resources
IR – Integration Resources

DWR – Design Work Rate
IWR – Integration Work Rate
t_{total} – Time required to complete the project
DP – Design Productivity
IP – Integration Productivity

And:

$$DW = \int_{0}^{t_{design}} (DWR)dt = \int_{0}^{t_{design}} [(DR)\cdot(DP)]\cdot dt \tag{4.1}$$

$$IW = \int_{0}^{t_{total}} (IWR)dt = \int_{0}^{t_{design}} (IWR(t))\cdot dt + \int_{t_{design}}^{t_{total}} [(IR)\cdot(IP)]\cdot dt \tag{4.2}$$

Where the integration work rate is a linear function with slope proportional to the maximum integration work rate and the time required to finalize the design work:

$$IWR(t) = \frac{(IR)\cdot(IP)}{t_{design}}\cdot t \tag{4.3}$$

After solving the integrals above, the time required to complete the project (units represented in brackets) is:

$$t_{design}[weeks] = \frac{DW[tasks]}{(DR[people])\cdot\left(DP\left[\frac{tasks}{people\cdot week}\right]\right)} \tag{4.4}$$

$$t_{total}[weeks] = \frac{IW[tasks]}{(IR[people])\cdot\left(IP\left[\frac{tasks}{people\cdot week}\right]\right)} + \frac{1}{2}\cdot\frac{DW[tasks]}{(DR[people])\cdot\left(DP\left[\frac{tasks}{people\cdot week}\right]\right)} \tag{4.5}$$

The equation 4.5 above confirms the prior results. When considering the total time of the project, managerial actions affecting the design work will have half the impact a similar action could have in the integration work. The final decision on how many people will be working in the project will have to be traded off with the relative cost of each alternative, since working overtime or hiring additional personnel will have different economic consequences.

To further develop the framework above for dependant tasks, the effect of agreeing on having a relative completion level for the design work in order to allow the integration work to be correctly managed was derived and the resulting equation modifies the equation above and is reproduced below in equation 4.6. Product commonality has a relevant effect in the total number of design and integration tasks, however, it will be addressed later throughout the complete system dynamics simulation.

$$t_{total}[weeks] = \frac{IW[tasks]}{(IR[people]) \cdot \left(IP\left[\frac{tasks}{people \cdot week}\right]\right)} + \frac{1}{2} \cdot \alpha \cdot \frac{DW[tasks]}{(DR[people]) \cdot \left(DP\left[\frac{tasks}{people \cdot week}\right]\right)} \quad (4.6)$$

Where:

α – Agreed design work completion % to adequately perform the integration work. An 80% completion threshold will be used following Pareto's rule.

The addition of α to the total time equation (4.6) brings the complete "iron triangle" framework in place, where the project's cost (represented by the number of people), schedule (represented by the total time) and scope (represented by α) are in tension to each other. A lower scope will improve time (and therefore resources); more people will reduce the time, but will increase the cost. No single solution is better than another, but the key insight to the project manager is to understand that the project's performance is a matter of tradeoff and not a matter of optimization of all the three sides of the iron triangle. Nevertheless, as concluded from equation 4.6, acting first on the integration resources could yield a faster completion compared to the design workstream.

A similar design and integration differentiation for the tasks and resources was found in the case study and was decided to be included in the model structure because both activities are affected differently by product commonality. Also, the company under study clearly differentiated both workstreams and resources in their organization, as they usually report to different areas. Furthermore, the sequential relationship between the tasks was also present in the case study. The model will be further expanded in the future sections differentiating the design and the integration work, in the meantime, the conceptual behavior of the sequential relationship of these tasks has been developed for future insights in a complete system dynamics model.

While the framework above brings some insights, these should be only applicable for tasks that are dependent to each other. It must be acknowledged that there are also some other integration tasks that may not be dependent on the design tasks. Eppinger et al. (1994) developed a framework which differentiates three types of tasks: dependant tasks (series), independent tasks (parallel) or interdependent tasks (coupled); these are represented in figure 4.8. While the first two are well understood and vastly managed with the critical path method and Gantt charts, the last relationship needs more advanced tools for their management like the proposed Design Structure Matrix or System Dynamics (de Weck, Lyneis, 2009).

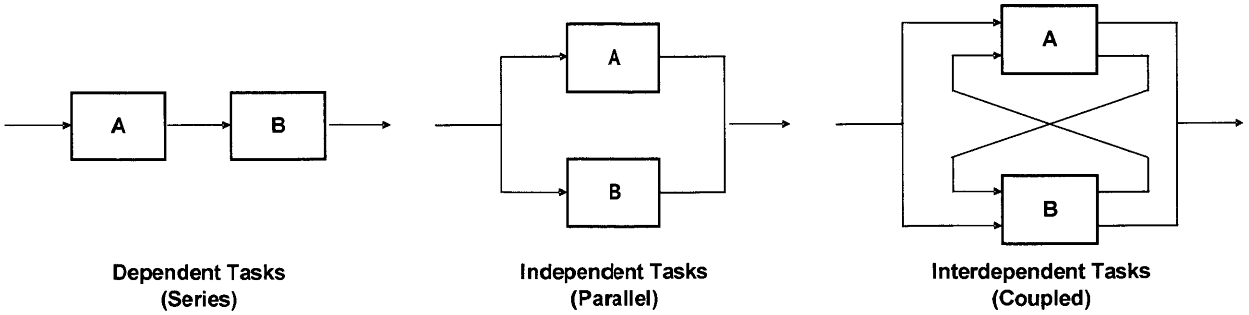


Figure 4.8: Characterization of tasks dependency and sequencing (Reproduced from Eppinger, et. al 2004)

If the tasks are dependent, then equation (4.6) is useful; however, if they are independent, then these tasks can be performed by each of their correspondent workforces separately. To correctly model the project, then the nature of the tasks (dependent, independent or interdependent) should also be included. The system dynamics model should account for the existence of the rework cycle, however, and the framework above needs to be completed with the independent tasks. If β represents the fraction of tasks that are independent from the design phase to the integration phase, and (β-1) represents the fraction of dependent tasks between both phases. The total time of the project (equation 4.7) will be derived by integrating figure 4.9 below:

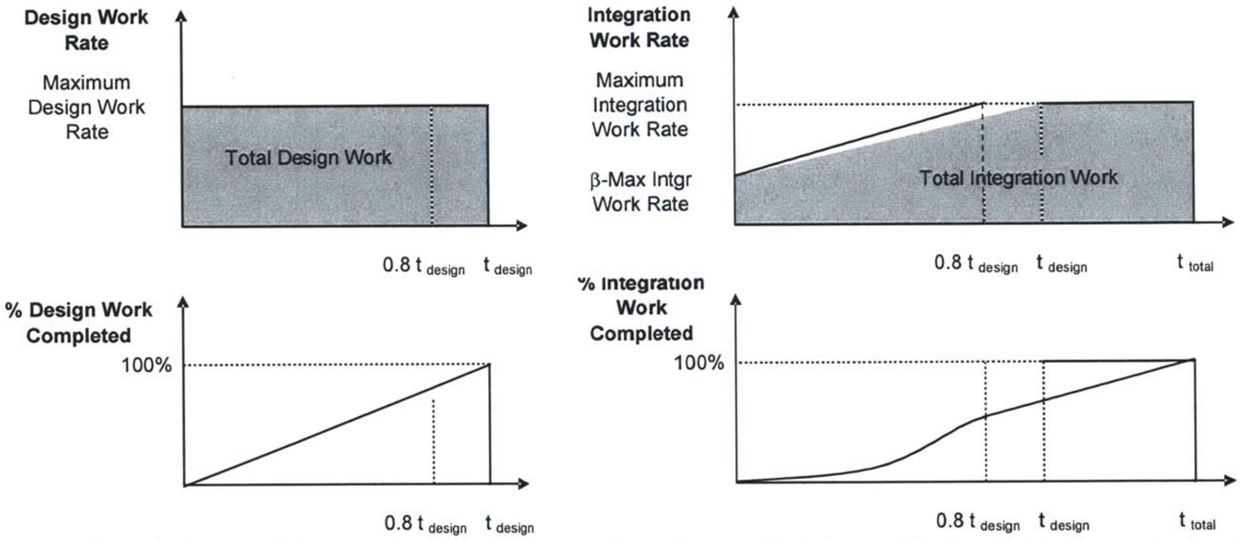


Figure 4.9: Design and integration work rate and work completed considering both dependent and interdependent tasks.

$$t_{total}[weeks] = \frac{IW[tasks]}{(IR[people])\cdot\left(IP\left[\frac{tasks}{people\cdot week}\right]\right)} + \frac{1}{2}\cdot\alpha\cdot(1-\beta)\frac{DW[tasks]}{(DR[people])\cdot\left(DP\left[\frac{tasks}{people\cdot week}\right]\right)} \tag{4.7}$$

From the equation above (4.7), if all the tasks were independent from the design phase to the integration, β would equal 1 and the total time would be the maximum between the integration and the design time. However if there is at least one dependent task within phases (β < 1), the integration tasks will not be complete until all the design tasks are completed and the equation above describes the total time. Similar to the prior equations, any managerial action taken to the integration tasks would directly impact the total project time; nevertheless, if a similar action is taken on the design resources, the improvement would be just a fraction of it, depending on the completion threshold (α) and the fraction of independent tasks (β).

4.4 Headcount Estimation and Incorporation into the Model

The primary objective of the system dynamics model is to understand how product commonality and divergence affect the development of the different derivative projects of a product family. Product commonality has a significant impact to the total number of people working in a project as found in the literature since common parts should yield more efficient product development (Meyer & Lenherd, 1997; Robertson and Ulrich, 1998; Muffatto, 1999).

Using the parts sharing studies developed in the prior section and the actual headcount required for the project, a general relationship between the number of common and unique parts was investigated. Table 4.2 summarizes the workforce required (in people-years) to complete the different projects, more detailed information was available for the first wave projects given the more advanced stage of their development.

	Egyptian Dog	Retriever	Greek Dog	Mayan Dog	Total Wave 1	Total Wave 2
Top Hat Resources	109.7	97.1	142.1	2.8	351.6	240.3
Platform Resources	146.5	24.2	64.0	0.0	234.7	257.6
Integration Resources	181.7	65.8	160.5	5.6	413.7	480.6
Total	*437.9*	*187.1*	*366.6*	*8.4*	***1000.0***	*978.5*

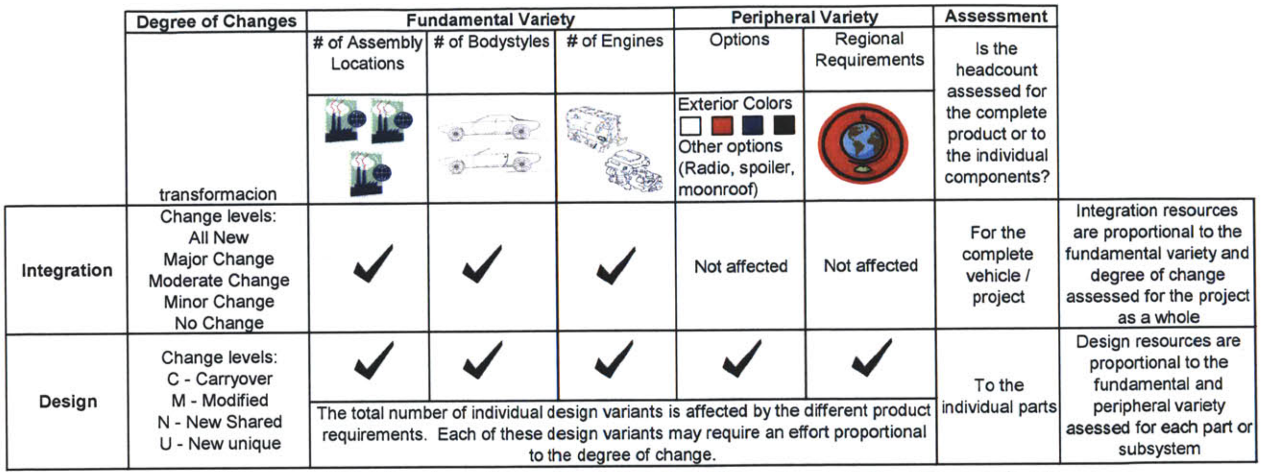
Table 4.2: Wave 1 and Wave 2 approved project headcount. Numbers are normalized to 1000 [people-year] for the total workforce required for wave 1.

The wave 1 headcount information above was assessed by the end of the underbody detailed design phase and all the information in the system dynamics simulation model will be correlated to its corresponding design commonality status by that time, corresponding to the first assessment of the parts sharing studies (see Figure 3.15). Similarly, the headcount information for the wave 2 was assessed by the end of the detailed design phase, although, it is reproduced for comparison purposes only. The workforce required to develop the engine and the transmission are not accounted in table 4.2 as they are usually managed outside the vehicle projects in the company.

A second characteristic about the workforce information above is that it does not correspond to the total workforce the project might need, but only the internal workforce for the company. If some components are designed by the supply base, the workforce is not included in the table above. If the components designed by the supply base were included in the project, it would only increase the number of the design resources, which currently fluctuate around 52% (wave 2) and 59% (wave 1). Clark and Fujimoto (1991) compared several automotive projects in the United States, Japan and Europe; and concluded that if all the parts were newly designed, the average integration workforce would represent 30% and the design workforce would represent 70% of the total project workforce. The workforce outside the project will not be included in the model that will only use the internal workforce information for its calibration.

The overall workload (or engineering hours) for an automotive project has been found to be directly related with the relative scope of the change (ratio of new to carryover parts) and the number of bodystyles and engines (Cusumano and Nobeoka, 1998; Clark and Fujimoto, 1991). Furthermore, Clark and Fujimoto differentiated the factors that contribute to the overall parts complexity to generate product variety. They found that "fundamental variety" significantly increases the engineering hours (i.e. number of bodystyles, number of engines and left hand and right hand drive configurations), while "peripherial variety" to the incremental options may not increase the total engineering hours significantly (i.e. exterior colors, special decals, or additional chrome).

The way the headcount is assigned in the projects under study is also directly correlated with the relative degree of change to the product and the relative complexity and scope of the project (mainly the number of engines and bodystyles), considering product commonality. For the integration activities, the headcount is proportional to the fundamental variety: number of assembly plants, number of bodystyles, number of engines and transmissions and the relative degree of change to the prior product for the exterior and interior, the chassis and the powertrain. For the design activities, on the other hand, it is proportional to the relative degree of change using the C,M,N,U part classification logic assessed for each of the parts and the fundamental and peripheral variety that affect directly the number of design variations. The Company's headcount drivers are summarized in figure 4.10.



	Degree of Changes	Fundamental Variety			Peripheral Variety		Assessment	
		# of Assembly Locations	# of Bodystyles	# of Engines	Options	Regional Requirements	Is the headcount assessed for the complete product or to the individual components?	
	transformacion				Exterior Colors ☐ 🟥 🟦 ⬛ Other options (Radio, spoiler, moonroof)			
Integration	Change levels: All New Major Change Moderate Change Minor Change No Change	✓	✓	✓	Not affected	Not affected	For the complete vehicle / project	Integration resources are proportional to the fundamental variety and degree of change assessed for the project as a whole
Design	Change levels: C - Carryover M - Modified N - New Shared U - New unique	✓	✓	✓	✓	✓	To the individual parts	Design resources are proportional to the fundamental and peripheral variety asessed for each part or subsystem
		The total number of individual design variants is affected by the different product requirements. Each of these design variants may require an effort proportional to the degree of change.						

Figure 4.10: Factors that affect the total headcount for the project.

As seen in figure 4.10 above, headcount depends on several factors; however, if the designs are common for the different derivatives, then it should reduce the required headcount when considering the workload for more than one vehicle and avoid counting the common parts twice. Since most of the parts were new for the vehicles, the total number of parts required for the set of projects can be a good approximation to the total headcount as assessed in the company. Using the parts sharing studies and the headcount per project, a strong correlation was found between the total number of parts and the headcount by comparing two, three or all the four projects from the first wave. Only the two of the logical combinations were excluded from this correlation as they did not follow the logic for the total amount of work as the Mayan Dog project was only comparable when it included the common parts from both the Egyptian and the Greek Dog projects. Table 4.3 and figure 4.11 summarize these results.

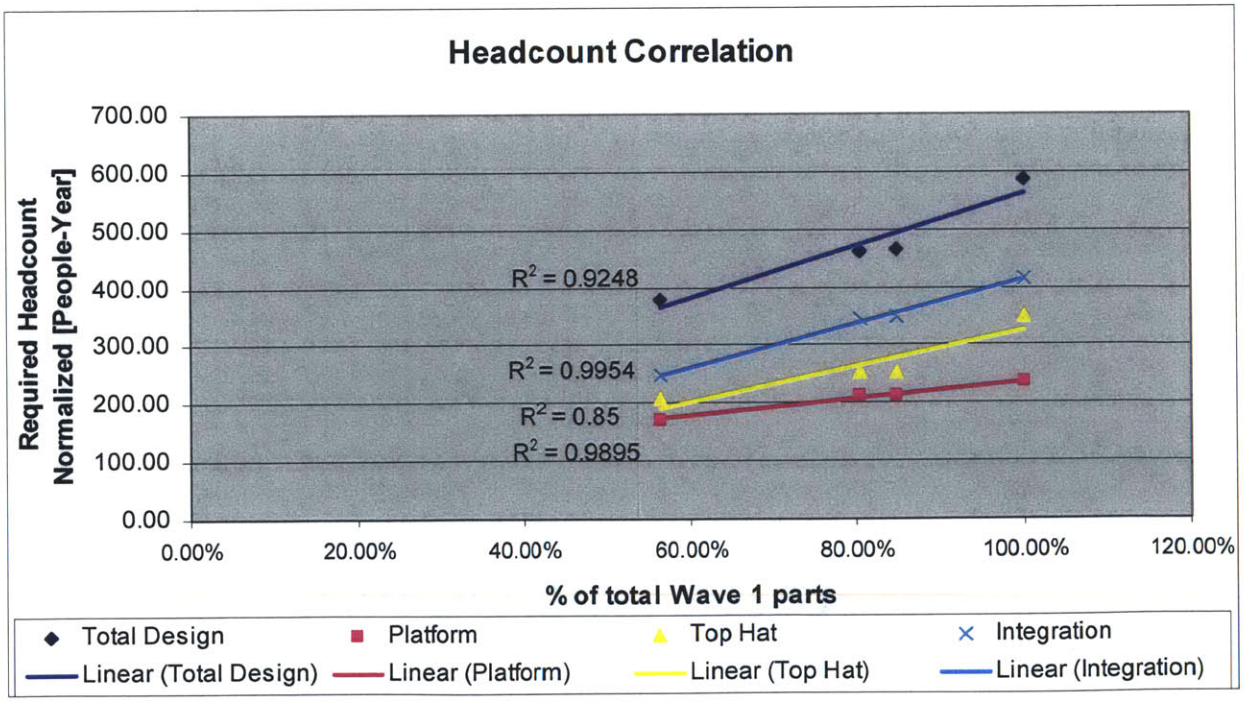


Figure 4.11. Headcount correlation when comparing two or three projects.

	% of Wave 1 Parts	Total Design	Platform	Top Hat	Integration
Egyptian Dog & Retriever	56.47%	377.47	170.71	206.76	247.57
Egyptian Dog & Greek Dog	80.45%	462.24	210.45	251.79	342.25
Egyptian/Greek Dog & Mayan Dog	84.81%	465.04	210.45	254.59	347.85
All Wave 1 (Dog)	100.00%	586.32	234.67	351.64	413.68
Egyptian Dog & Mayan Dog	54.48%	258.98	146.48	112.50	187.33
Greek Dog & Mayan Dog	71.09%	208.85	63.97	144.88	166.11

Table 4.3: Correlation between the number of parts and the headcount.

Given the strong correlation between headcount and the total number of parts (common or unique), the headcount required was assumed to be proportional to the total number of parts required:

- Integration resources: 586 people-year per each 1% of the total wave 1 parts
- Top Hat resources: 234 people-year per each 1% of the total wave 1 parts
- Platform resources: 351 people-year per each 1% of the total wave 1 parts

After determining the relationship between product commonality and total required headcount for the project, the second headcount characteristic that was included in the model was the headcount curve throughout the project which is common in product development projects, characterized for a ramp up period and a ramp down period. This headcount curve is based on the organization's product development system and historical performance of prior product development projects.

The company's specific headcount curve peaks to a maximum when the vehicle design is completed. This is achieved when all the blueprints and mathematical models of all the parts are completed and quoted by the supply base. After the design is completed, the project goes through a major review and if approved, all the parts will be authorized for production and their respective production tools are authorized to start their development. Other workstreams that start after this authorization are the population of these parts in the company's production parts database and the construction of the first complete functional prototype. Figure 4.12 below details the project's descriptive timeline with the major design tasks and the curve detailing the headcount requirements to complete the project. The scale on the left represents the percentage of the maximum headcount required. The headcount curve was adjusted to the actual project headcount using a normal distribution.

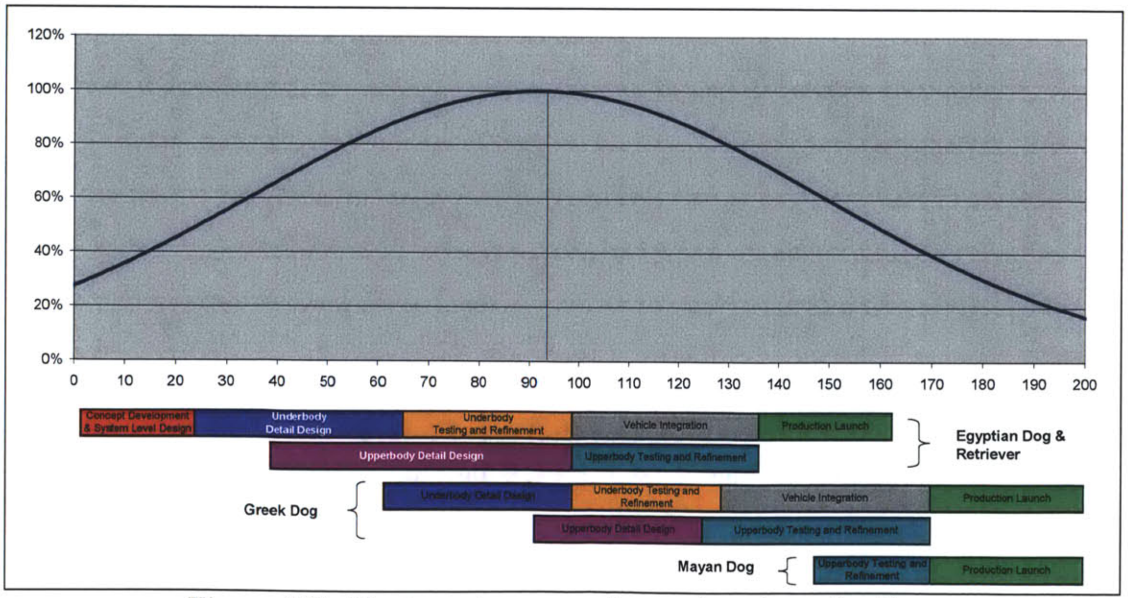


Figure 4.12: Headcount curve and wave 1 projects timeline.

Figure 4.12 above represents the total number of people compared to the maximum number of people working at any given time in the project. The headcount curve above was derived starting from the total engineering hours required, considering that the total engineering hours is represented by the area below the headcount curve assuming the shape of the curve was the proposed adjusted normal distribution. The equivalence conversion is shown in figure 4.13 which represents the total headcount required and how it correlates to the proposed headcount curve. If 1000 people-year were assumed over the project time, it should then represent a total of 250 people working during the entire project (left graph) or a maximum of 366 people during the peak period with the assumed headcount curve.

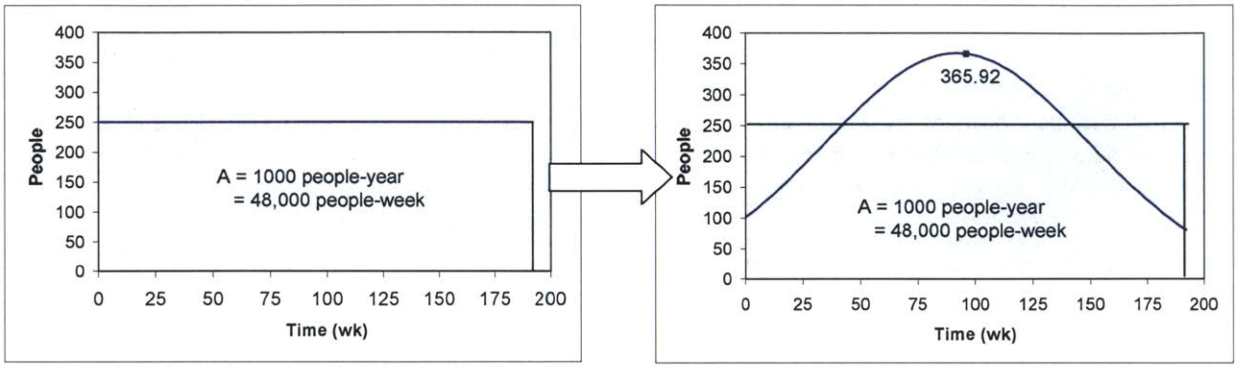


Figure 4.13: Correlation between the headcount required and the headcount curve

For the system dynamics model, the original generic dimensionless curve with 1 or 100% corresponding to the peak point will be used and will be scaled to the required time and headcount required for the projects under study.

4.5 Tasks Correlation and Preliminary Model Calibration

The system dynamics models for project management require understanding basic information about the atomic unit of the project: the task. A detailed work breakdown structure (WBS) can be found in the organization's product development system; however, all these tasks may have very different nature and therefore very different time to complete each of the tasks. The model requires knowing how many tasks (design tasks or integration tasks) are required to complete the project and also requires all these tasks to be standardized so each task has the same equivalent completion time.

To standardize the duration and number of tasks, instead of finding an average task time from the total tasks from the work breakdown structure available, an average task was created using the headcount defined in the prior section to standardize all the project's tasks. A predefined productivity of 1 task per person per week was assumed to define the length of the task. These two assumptions made it possible to estimate the number of tasks required to complete the project based on the headcount curve detailed above. To estimate the total number of tasks, we can calculate the average headcount integrating the headcount curve for the whole project time. The total number of tasks, based on the definition of productivity (number of tasks per people per unit of time), will be:

$$Productivity\left[\frac{Tasks}{People \cdot Week}\right] = \frac{Work\ to\ do\ [Tasks]}{Average\ Headcount\ [People] \cdot Project\ Duration\ [Weeks]} \tag{4.8}$$

However, because the average headcount varies over time, a more accurate representation will be the total workforce in People-Weeks:

$$Productivity\left[\frac{Tasks}{People \cdot Week}\right] = \frac{Work\ to\ do\ [Tasks]}{Required\ Workforce\ [Weeks]} \tag{4.9}$$

Therefore, the work to do will be represented by the following relationship:

$$Work\ to\ do\ [Tasks] = Productivity\left[\frac{Tasks}{People \cdot Week}\right] \cdot Required\ Workforce\ [People \cdot Weeks] \tag{4.10}$$

To estimate the required workforce, the project's headcount curve will be required. In this case, the workforce will be obtained by integrating the headcount over a predefined period of time (the project duration). The relationship for the required workforce as a function of time will be an S-shaped curve. Figure 4.14 below explains graphically the correlation between the required workforce and the headcount.

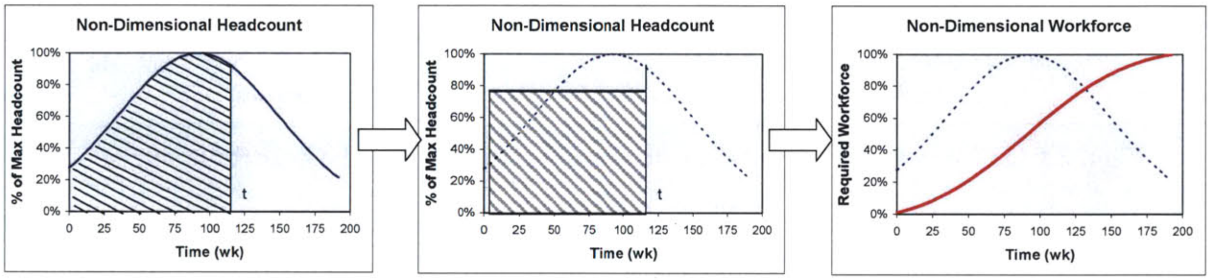


Figure 4.14: Relationship between the non-dimensional headcount and the required workforce.

The required workforce curve is valuable for estimating the total tasks, as it indicates the total required workforce for given a pre-defined project duration, as determined by the product development system of the company. The simulation will account for the project behavior for different cut-off dates concurrent with the design stages: conceptual design, detailed design and testing and refinement. Therefore, figure 4.15 reproduces the number of tasks as a percentage of the total project tasks corresponding to the different design stages, where 100% represents the total tasks required for the project, corresponding to the predefined project time for the lead derivative.

Using the headcount and tasks as defined, the system dynamics model will be correlated with these parameters. Four groups of projects were selected to be correlated with the simulation, based on the logical bundling of these projects inside the company:

- Egyptian Dog and Retriever
- Egyptian Dog and Greek Dog
- Egyptian/Greek Dog and Mayan Dog
- All Wave 1 Projects: Egyptian Dog/Retriever and Greek/Mayan Dog

These four cases will be simulated in two different scenarios: with and without the ramp-up curve. These simple cases will be further compared with future model features like product commonality, design divergence, lifecycle offsets and rework. Table 4.4 details the input parameters for these simple simulations that will not include the effect for the headcount curve deployment.

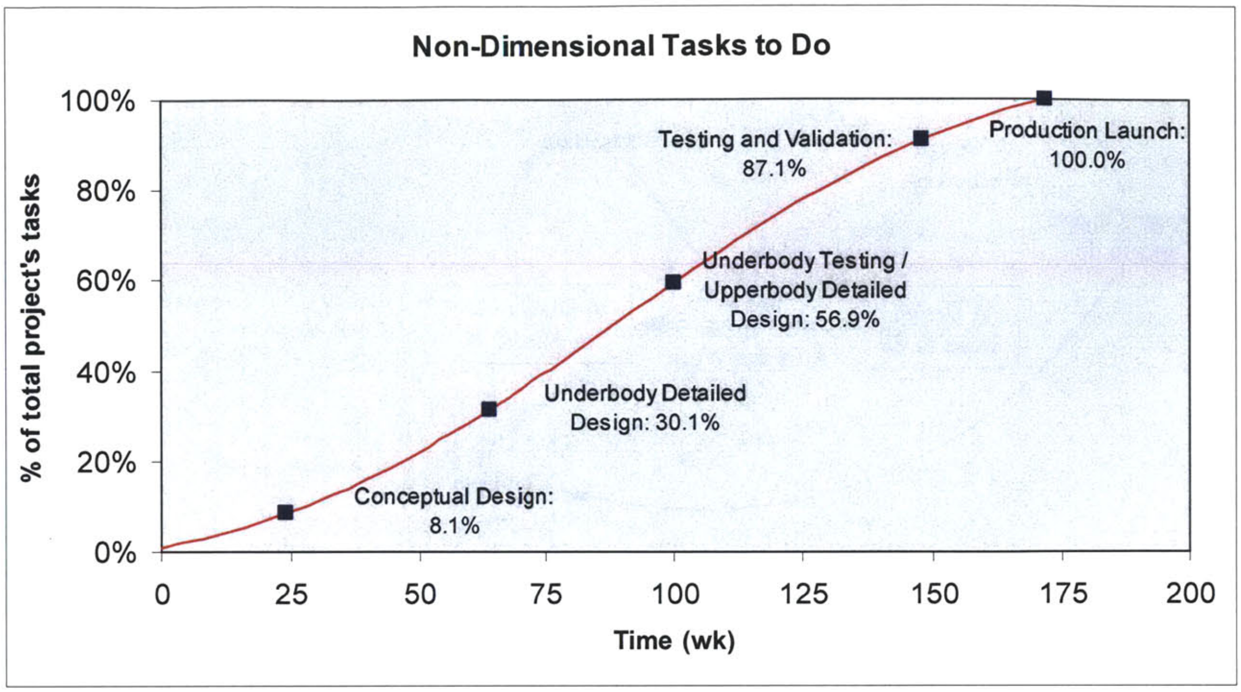


Figure 4.15: Non Dimensional tasks to do, representing the percentage of the total tasks to complete the project.

Case Study	Project A	Project B	Commonality [%] of Total Wave 1 Projects			Required Workforce [People-Week]		Required Headcount [People]		Work To Do [Tasks]	
			Unique Parts A	Common Parts	Unique Parts B	Design	Integration	Design	Integration	Design	Integration
1	Egyptian Dog	Retriever	25.0%	15.3%	16.2%	15891.9	11212.6	92.4	65.2	15891.9	11212.6
2	Egyptian Dog	Greek Dog	15.3%	25.0%	40.2%	22640.6	15974.1	131.6	92.9	22640.6	15974.1
3	Egyptian and Greek Dog	Mayan Dog	49.9%	30.6%	4.3%	23867.0	16839.5	138.8	97.9	23867.0	16839.5
4	Egyptian dog and Retriever	Mayan and Greek Dog	28.9%	27.6%	43.5%	28143.4	19856.6	163.6	115.4	28143.4	19856.6

Table 4.4: Case studies parameters definition without headcount curve

Figure 4.16 represents the simplified system dynamics model without the headcount curve. The model below was used to simulate the projects without some future model additions like the headcount curve, rework or divergence. The three case studies above will be simulated with this simple setup so the baseline model behavior will be understood a priori. The project dynamics behave as indicated in equation 4.7. Some of the simulation results are shown in figures 4.17 and 4.18 which represent the tasks completed and the integration work rate, a similar behavior was found for the other projects, as expected. The results of all the simulation cases and runs and cases where compared to the expected results from equation 4.7 are summarized in Table 4.5, which also details the sensitivity to the parameters α (the completion threshold or acceptance level) and β (the fraction of independent tasks from the design to the integration phase in the project) to the total project time.

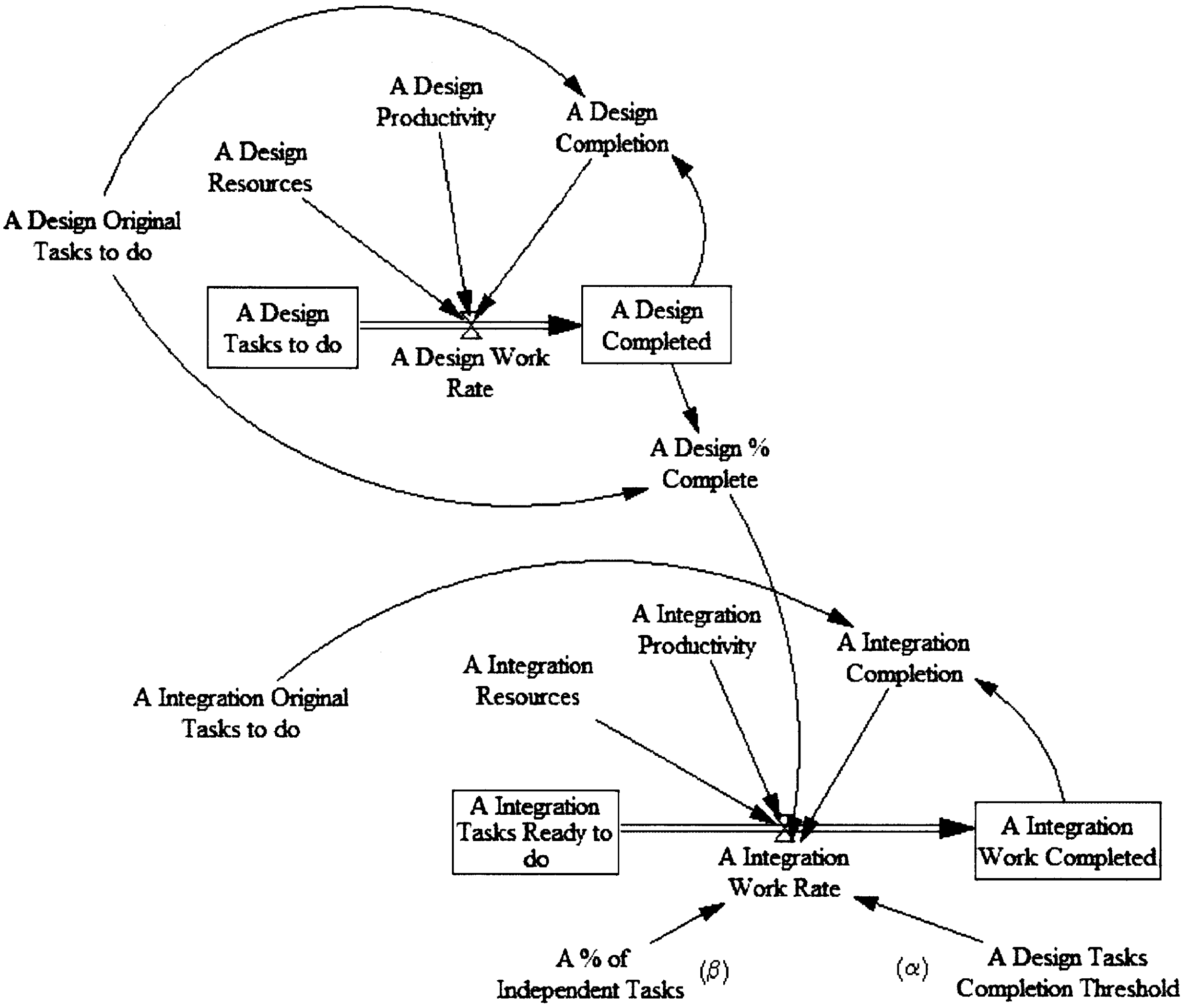


Figure 4.16: Project's simplified system dynamics model without rework and without the headcount curve.

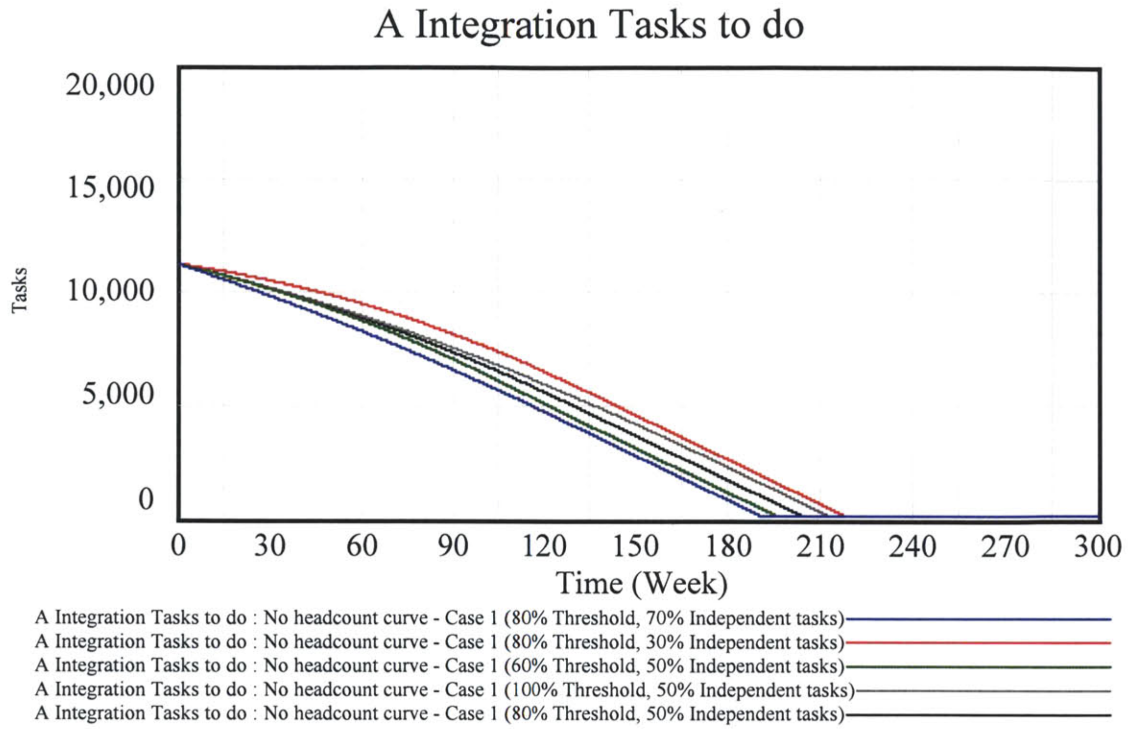


Figure 4.17: Case 1 (Egyptian Dog vs. Retriever) simulation results. The project ends when all the integration tasks are completed.

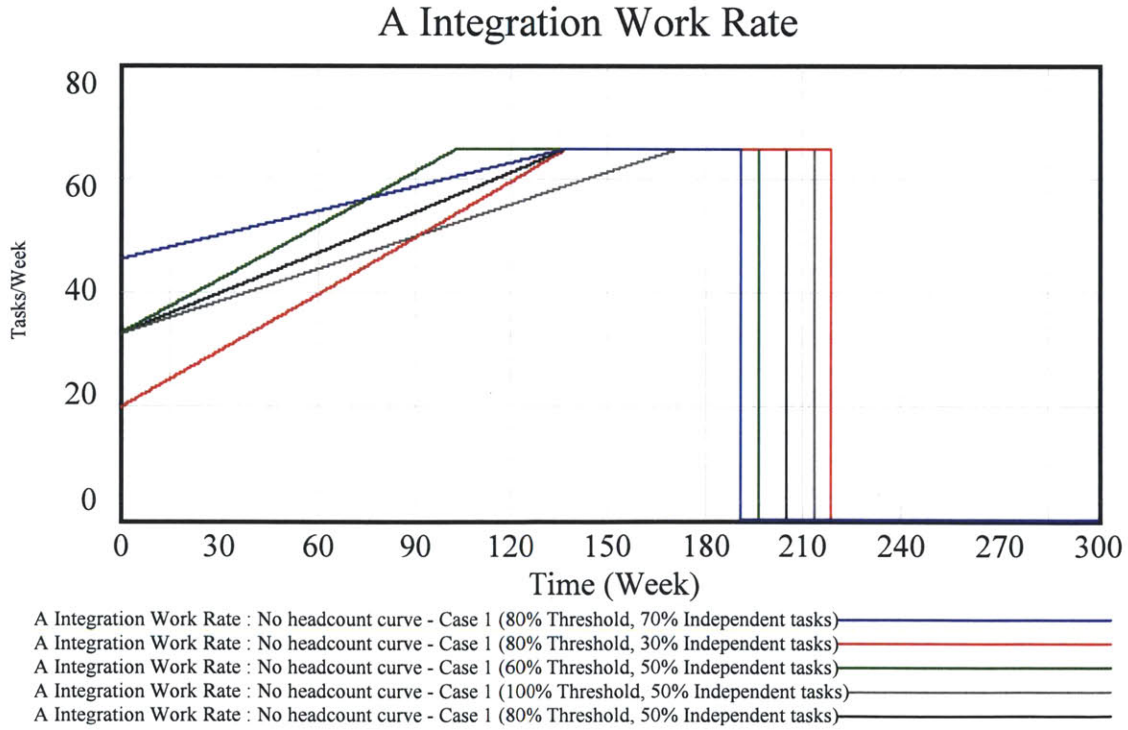


Figure 4.18: Case 1 (Egyptian Dog vs. Retriever) simulation results. The integration work rate was modified with the α and β parameters.

Case Study	α - Completion Threshold	β - % of Independent Integration tasks	Expected Project Time (if all tasks were independent) [Tasks]	Simulated Total Project Time [Weeks]	Estimated Total Project Time (Eq 4.7) [Weeks]	Incremental time due to dependant tasks [%]
1	80%	50%	172	206.43	206.4	20.0%
	100%	50%		215.06	215	25.0%
	60%	50%		197.81	197.8	15.0%
	80%	30%		220.18	220.16	28.0%
	80%	70%		192.68	192.64	12.0%
2	80%	50%	172	206.43	206.4	20.0%
	100%	50%		215.06	215	25.0%
	60%	50%		197.81	197.8	15.0%
	80%	30%		220.18	220.16	28.0%
	80%	70%		192.68	192.64	12.0%
3	80%	50%	172	206.43	206.4	20.0%
	100%	50%		215.06	215	25.0%
	60%	50%		197.81	197.8	15.0%
	80%	30%		220.18	220.16	28.0%
	80%	70%		192.68	192.64	12.0%
4	80%	50%	172	206.43	206.4	20.0%
	100%	50%		215.06	215	25.0%
	60%	50%		197.81	197.8	15.0%
	80%	30%		220.18	220.16	28.0%
	80%	70%		192.68	192.64	12.0%

Table 4.5: System dynamics model simulation results compared to equation 4.7.

The results in table 4.5 confirm the validity of the model and equation 4.7 as they correlate perfectly. The expected project time is the same for all four case studies, as the headcount and the tasks are scaled to the project content based on their commonality. The project's completion time is in general more sensitive to the fraction of independent tasks than the completion threshold. These parameters will be left as 80% as the completion threshold and 50% for the fraction of independent tasks for future simulations, understanding that these parameters yield a project slip time of 20% that has to be managed or adjusted with a different parameter, like the integration productivity.

The prior model assumes incorrectly that the headcount is constant throughout the project, so the next step is to update the model to include the effect of the headcount curve and estimate how much is the project time slip for each of the design phases. Table 4.6 below details the different inputs for the four case studies. The headcount will be modeled using the non-dimensional headcount curve adapted for to the expected project duration of 172 weeks. On the other hand, the number of tasks will be derived from the headcount curve and the expected project completion fraction as detailed previously in figure 4.12.

Case Study	Project A	Project B	Design Stage	Required Workforce [People-Week]		Peak Resources [People]		Work To Do [Tasks]	
				Design	Integration	Design	Integration	Design	Integration
1	Egyptian Dog	Retriever	Concept Development	1347.6	950.8	126.8	89.5	1347.6	950.8
			Detailed Design	4991.6	3521.9	126.8	89.5	4991.6	3521.9
			Testing and Refinement	9441.4	6661.4	126.8	89.5	9441.4	6661.4
			Complete Project	15891.9	11212.6	126.8	89.5	15891.9	11212.6
2	Egyptian Dog	Greek Dog	Concept Development	1919.9	1354.6	180.7	127.5	1919.9	1354.6
			Detailed Design	7111.4	5017.5	180.7	127.5	7111.4	5017.5
			Testing and Refinement	13450.8	9490.2	180.7	127.5	13450.8	9490.2
			Complete Project	22640.6	15974.1	180.7	127.5	22640.6	15974.1
3	Egyptian and Greek Dog	Mayan Dog	Concept Development	2023.9	1428.0	190.4	134.4	2023.9	1428.0
			Detailed Design	7496.6	5289.3	190.4	134.4	7496.6	5289.3
			Testing and Refinement	14179.4	10004.3	190.4	134.4	14179.4	10004.3
			Complete Project	23867.0	16839.5	190.4	134.4	23867.0	16839.5
4	Egyptian dog and Retriever	Mayan and Greek Dog	Concept Development	2386.6	1683.8	224.6	158.4	2386.6	1683.8
			Detailed Design	8839.8	6237.0	224.6	158.4	8839.8	6237.0
			Testing and Refinement	16720.0	11796.8	224.6	158.4	16720.0	11796.8
			Complete Project	28143.4	19856.6	224.6	158.4	28143.4	19856.6

Table 4.6: Case studies parameters definition with the headcount curve

The prior simulations and equation 4.7 set the expectations for the expected total project time in an ideal situation where the headcount is constant throughout the project given the parameters α and β. However, the project's headcount curve is based on a fixed assumption that the project will be actually completed on the expected time. When the headcount curve is included in the model, the simulation requires some adjustments to the model because while the design tasks are completed as expected (since the headcount curve was adjusted to do so), the integration tasks require either more time, more people or a lower completion threshold to be completed in the expected time. The effect in the simulation is a long tail for the headcount curve that requires significantly more time to complete the project. This effect is shown in figures 4.19 and 4.20 below.

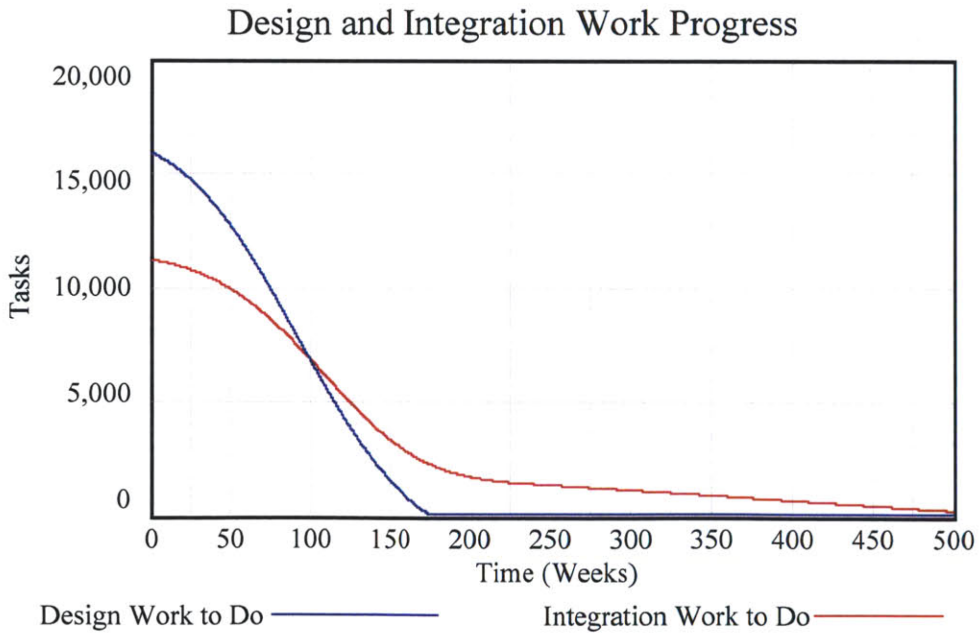


Figure 4.19: Work progress for the design and the integration activities. The integration activities will have a long tail if the resources curve is the same as the design activities.

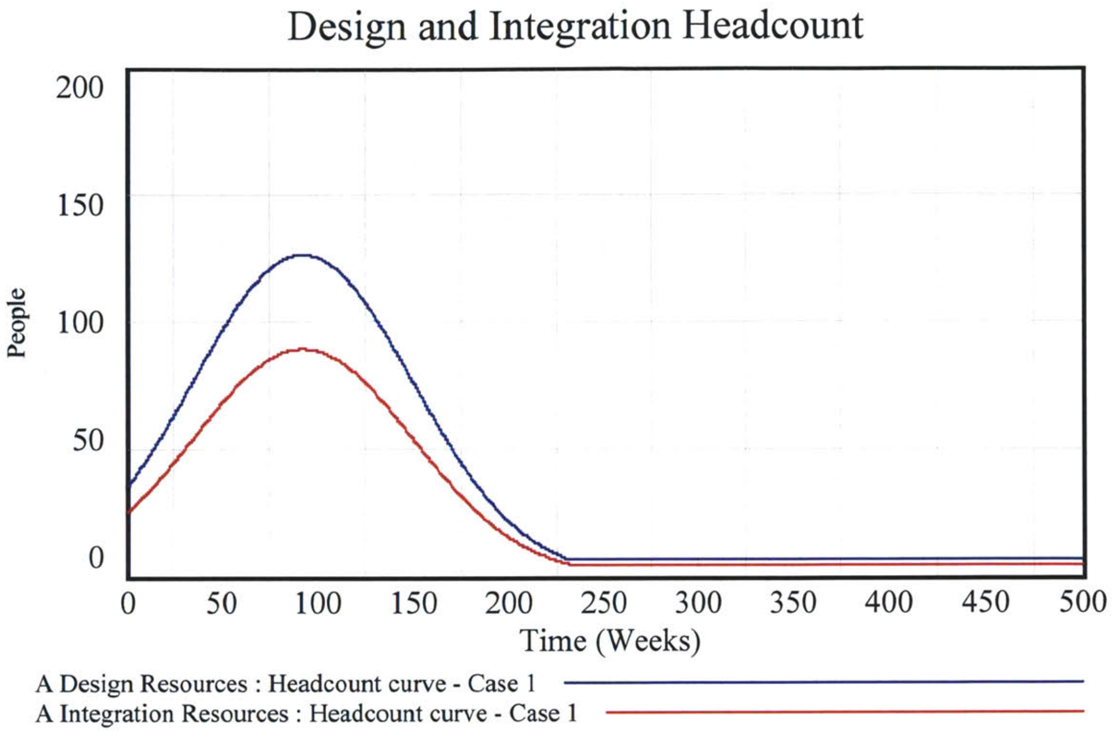


Figure 4.20: Headcount curve effects for the design and the integration activities. The tail was defined as 5% of the peak point.

To resolve the situation above, which represents a real problem in product development projects, one common solution is to "freeze" the design to an agreed date or engineering level to allow the integration activities to perform their activities and perform a point in time assessment. The headcount curve will therefore be adjusted to assume all the project should end before the indicated date for the design activities and on the indicated date for the integration activities. The same headcount shape will be assumed for both, however, the peak point will be shifted between both curves by this indicated offset for completing the design activities. Figure 4.21 illustrates the new headcount curves and shows the offset between the integration and the design activities. The max headcount point is maintained, but the deployment between design and integration has an offset of 1 month. The offset between these curves will be called "headcount offset (γ)".

The adjusted headcount curves with a one month offset between them did not change the project behavior significantly compared to figure 4.19, however, the new setup will be left in the model as it captures the offset between headcounts as an additional parameter to adjust the model behavior. Table 4.7 summarizes the simulation results for the total project time for each of the different deadlines, compared to their expected time without the headcount curve.

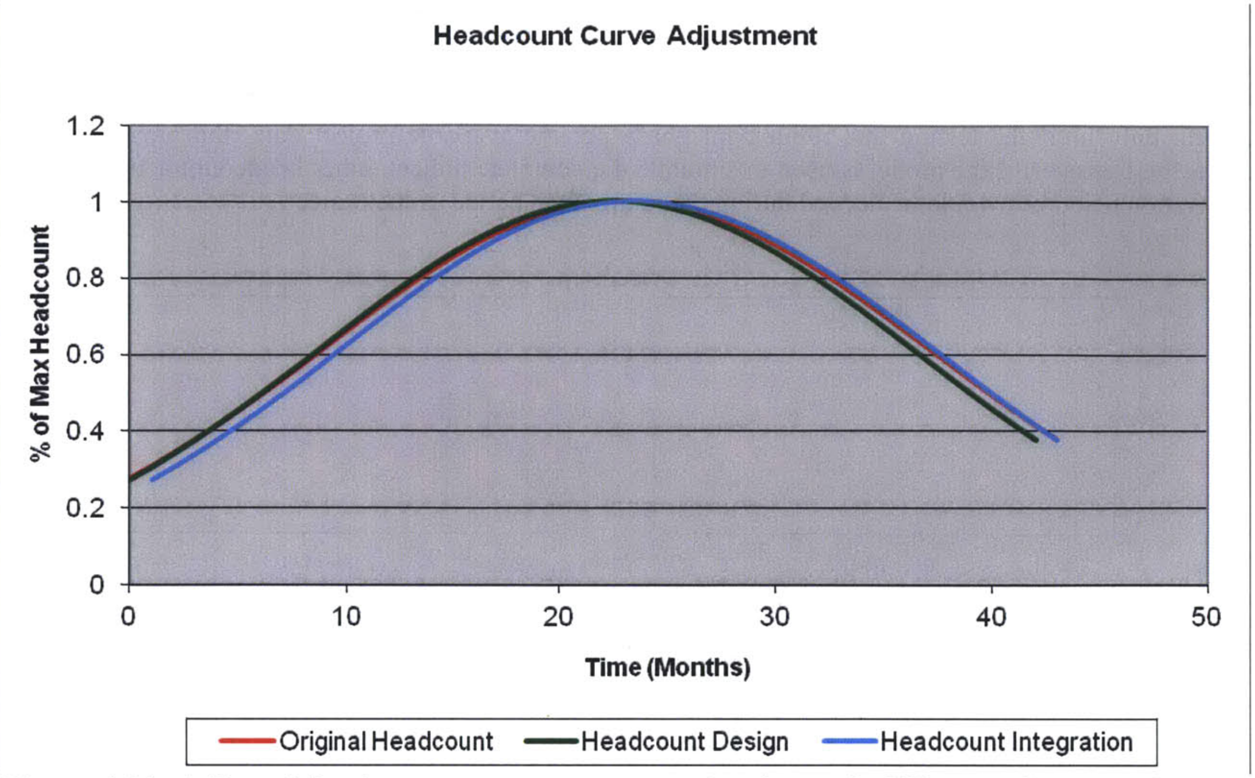


Figure 4.21: Adjusted headcount curves to account for 1 month difference between the peak headcount points.

			Headcount Scenario								
			Constant Headcount with design and integration workstreams: α(Completion threshold)=0.8, β(% of Independent tasks)=0.5			Variable Headcount - Time Required					
Case Study	Design Stage	Planned Completion Time [Week]				Peak Resources [People]		Concurrent headcount curve [Week]	Headcount curve with offset		
			Avg Design Headcount [People]	Avg Integ Headcount [People]	Time Required [Week]	Design [People]	Integration [People]		Offset: 4 wk	Offset: 8 wk	Offset: 12 wk
1	Concept Development	24.0	56.2	39.6	28.8	126.8	89.5	30.75	32.31	34.25	36.25
	Detailed Design	64.0	78.0	55.0	76.8			74.625	76.31	78.69	81.19
	Testing and Refinement	100.0	94.4	66.6	120			117.44	118.81	120.75	122.81
	Complete Project	172.0	92.4	65.2	206.4			> 500	> 500	499.25	464.50
2	Concept Development	24.0	80.0	56.4	28.8	180.7	127.5	30.75	32.31	34.25	36.25
	Detailed Design	64.0	111.1	78.4	76.8			74.625	76.31	78.69	81.19
	Testing and Refinement	100.0	134.5	94.9	120			117.44	118.81	120.75	122.81
	Complete Project	172.0	131.6	92.9	206.4			> 500	> 500	499.25	464.18
3	Concept Development	24.0	84.3	59.5	28.8	190.4	134.4	30.75	32.31	34.25	36.25
	Detailed Design	64.0	117.1	82.6	76.8			74.625	76.31	78.69	81.19
	Testing and Refinement	100.0	141.8	100.0	120			117.44	118.81	120.75	122.81
	Complete Project	172.0	138.8	97.9	206.4			> 500	> 500	499.62	464.56
4	Concept Development	24.0	99.4	70.2	28.8	224.6	158.4	30.75	32.31	34.25	36.25
	Detailed Design	64.0	138.1	97.5	76.8			74.625	76.31	78.69	81.19
	Testing and Refinement	100.0	167.2	118.0	120			117.44	118.81	120.75	122.81
	Complete Project	172.0	163.6	115.4	206.4			> 500	> 500	499.62	464.56

Table 4.7. Project duration simulation results considering constant headcount or the headcount curve with or without headcount peak offset between the design and integration headcount.

After comparing the simulation results, the headcount curve had a significant effect for the complete project. Yet, when comparing the project schedule with the different project reviews deadlines, the schedule deteriorates compared to the constant headcount scenario only for the concept development review, the other two check points result in a better schedule because the headcount curve is increasing significantly during those phases. This insight is illustrated in figure 4.22 that shows the work rate for the integration tasks and the work rate for the design tasks; the headcount offset and the large available workforce at the end of the testing and refinement phase is helping the project to complete the tasks, but it would create a tail if the complete project is simulated. Given that the simulation can be correlated to the project only until the testing and refinement phase per the available project data, the model behaves correctly, however, if the model is used to simulate the complete project, other headcount actions for the integration resources would be needed during the ramp down period of the project.

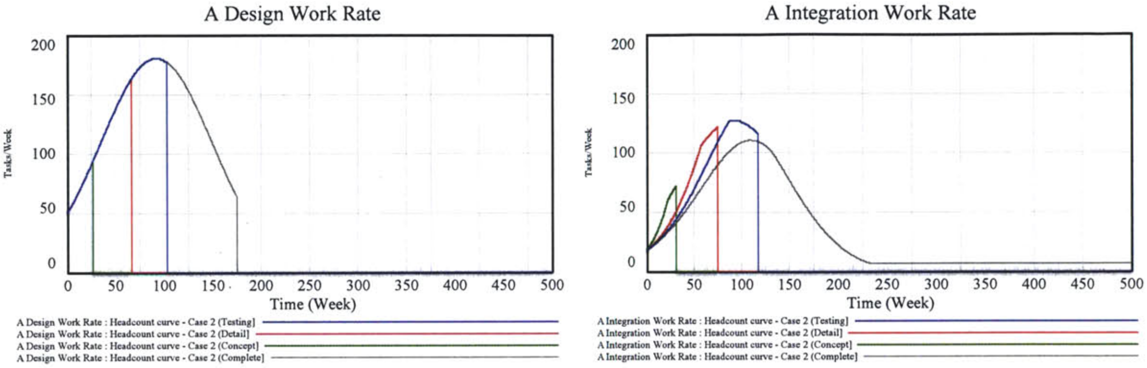


Figure 4.22. Design and integration work rate comparison for the four different cutoff points.

4.6 Incorporation of Quality and Rework into the Model

The key feature of the system dynamics models for projects compared to the conventional project management tools is the ability to incorporate rework into the model. On all the prior models, rework has not been assumed, however, the headcount has been derived from prior projects experience and therefore assumes that rework is present as part of the normal product development process. The rework magnitude will be controlled by a new variable named "Quality" that represents the fraction of the tasks that are completed correctly.

Figure 4.23 shows the updated system dynamics model that incorporates rework. Besides the new parameter "quality" a new stock named "undiscovered rework" is included in the model that corresponds to all the tasks that are not done correctly, initially unbeknownst, and yet that have to be reworked eventually. The "undiscovered rework" is affected by two different flows: the rework generation and the rework discovery. The first flow is proportional to "quality" or the tasks that are not done correctly and the second flow is affected by the speed of discovering rework by downstream activities like testing. Finally, the "time to discover rework" represents the average delay of finding errors in the tasks.

To incorporate "quality" and rework into the model, the constant headcount scenario was assumed to understand the sensitivity to the "quality" parameter for both the design and the integration activities. To calibrate the "quality" parameter, a full project simulation of case 1 (Egyptian Dog and Retriever) was performed, a similar behavior should be expected for the other four cases as they have been scaled up or down with their relative resources and tasks. The following inputs were assumed for the updated model with rework and constant headcount:

- α (Design completion threshold) = 80%
- β (% of independent tasks) = 50%
- Time to discover rework: 1 week for the integration resources and 4 weeks for the design resources. The time to discover rework for the design activities is consistent with the proposed "design freeze" timing of 1 month as discussed previously. On the other side, 1 week rework

discovery time proposal for the integration activities reflects the shorter time to deadlines and in some cases could be even a smaller time.

- Project completion: given the iterative nature of the simulation with rework, the project will be "finalized" if 99% of the integration tasks are completed. This is a common practice in these models to avoid infinite loops.
- In order to avoid staff to work even if there are no left tasks to be done, the model requires constraining the work rate by selecting the lowest work rate based on the available staff work rate potential or the work rate based on the tasks available based on the minimum time to perform a task that will be equal to 1 week.
- Same tasks and people as in table 4.4. Case study 1 was included in this simulation.

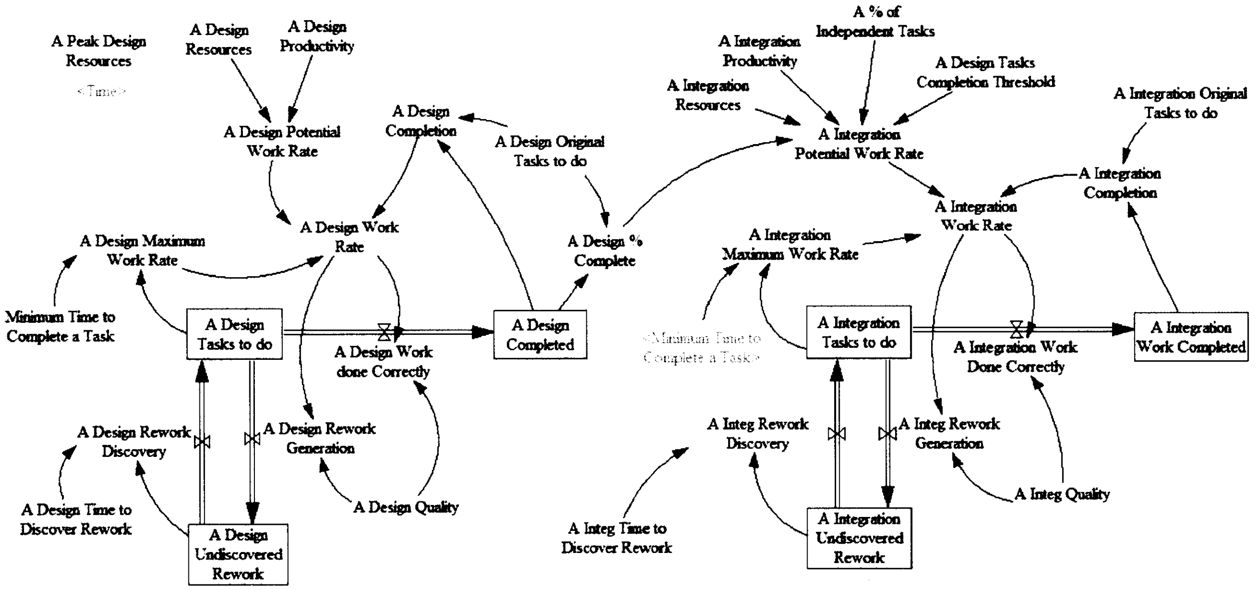


Figure 4.23: Updated model assuming rework for the design and the integration activities.

Table 4.8 compiles the results of the sensitivity analysis on the design and integration quality, where the percentage of schedule overrun is compared to the first model which already assessed a 20% overrun as the baseline. The results show that the project schedule is more sensitive to the integration quality. This behavior was expected since the design resources have much more capacity to overcome rework while the integration resources have to overcome both the design and the integration rework, as well as the pending tasks to be completed that have been delayed due to the design tasks completion. Also, if the tasks are not done correctly towards the end of the project, these will have to be reworked and the project schedule will slip inevitably without enough time to react.

Estimating the "quality" parameter for the simulation and correlating it to the actual project is a difficult task. As stated previously, the nature of all the individual tasks has a wide range, either for the design or integration tasks. Examples of tasks to be completed in the design projects are engineering drawings, 3D models, validation and verification tests, fabrication of prototypes, assessing the project profitability, etc. When the vehicle starts its production, the main task the design activities must deliver are the individual components to the assembly plant that must be fabricated with the expected quality. The integration tasks must deliver an affordable, functional and high quality vehicle overall. To achieve these main tasks, a significant amount of design changes should happen before the start of production.

Design Quality	Integration Quality	Project Time	% Schedule increase	Design Quality	Integration Quality	Project Time	% Schedule increase
100%	100%	204.62	0.00%	85%	100%	210.68	2.96%
	95%	213.63	4.40%		95%	219.68	7.36%
	90%	223.56	9.26%		90%	229.63	12.22%
	85%	234.69	14.70%		85%	240.75	17.66%
	80%	247.19	20.80%		80%	253.25	23.77%
95%	100%	206.43	0.88%	80%	100%	213.25	4.22%
	95%	215.43	5.28%		95%	222.19	8.59%
	90%	225.38	10.15%		90%	232.13	13.44%
	85%	236.5	15.58%		85%	243.25	18.88%
	80%	249	21.69%		80%	255.81	25.02%
90%	100%	208.43	1.86%				
	95%	217.43	6.26%				
	90%	227.38	11.12%				
	85%	238.5	16.56%				
	80%	251	22.67%				

Table 4.8: Design and integration quality sensitivity.

Instead of determining the average quality of a wide range of tasks, the iterative nature of product development and the design churn will be incorporated into the model. The model will monitor design completion (part drawings available) as the main task, and will be measured with the design completion of each of the parts of the BoM. "Quality" will be changed for the average churn (number of design changes) throughout the development of the project, as it is easier to measure and is easily understood.

4.6 Design Churn and Change Control Incorporation into the Model

The main source of information available for the projects under study was the change control meeting minutes. The nature of the change control meeting was previously described in section 3.4.3, along with the behavior of the meeting throughout the project. The first parameter to estimate in the model is the average number of design changes per part, for this task, two different sources of information yielded a similar result of an average of 5 design changes per part throughout the design cycle. First, the average churn was estimated based on the average piece cost of the parts from the four BoMs; an average number of changes was estimated based on the cost of the part, and the results are shown in table 4.9. The second correlation was performed directly from the BoM and the platform change control meeting minutes which were classified by the vehicle subsystem affected by the change. The change control meeting information was available until the end of the design phase, representing 57% of the project duration, therefore, the average churn was adjusted as illustrated in table 4.10.

	Costs Distribution					Estimated lifetime changes	
	Egyptian Dog	Greek Dog	Mayan Dog	Retriever	Average	Changes per part	Total Churn
Parts lower than 1 USD	30.4%	33.7%	29.1%	28.2%	30.5%	1	30.5%
Between 1-5 USD	31.0%	34.1%	31.7%	29.0%	31.6%	3	94.8%
Between 5 and 25	26.0%	22.0%	25.9%	27.6%	25.2%	8	201.5%
Between 25 and 100 USD	10.4%	8.7%	11.3%	12.1%	10.5%	15	158.0%
More than 100 USD	2.2%	1.4%	2.0%	3.1%	2.2%	8	17.2%

Total Churn	**502.0%**

Table 4.9: Estimated parts churn based on the estimated lifetime changes per part.

Vehicle Subsystem	Number of Times addressed in Change Control	Number of Parts Per Vehicle				
		Egyptian Dog	Greek Dog	Mayan Dog	Retriever	Average
Complete Vehicle	142	333	437	270	307	336.75
Body Structures	192	37	182	33	38	72.5
Suspension	78	50	53	38	51	48
Body and Security Electronics	66	10	15	13	15	13.25
EDS	60	62	30	37	33	40.5
Transmission and Clutch	60	15	16	13	13	14.25
Brakes and Pedals	54	33	34	30	34	32.75
Climate	47	12	13	12	11	12
Fuel	39	18	20	18	16	18
Cooling	36	24	21	26	38	27.25
Engine	30	18	9	9	15	12.75
Cold and Hot End Exhaust	29	8	12	8	8	9
Power Supply	26	3	4	4	4	3.75
Powertrain Mounts	21	15	5	8	6	8.5
Steering	20	4	3	3	3	3.25
AIS	15	10	6	9	7	8
Driveline	12	5	4	4	4	4.25
PCM	9	8	10	5	5	7
Chassis Electronics	5	1	0	0	6	1.75
Total	**941**	333	437	270	307	**336.75**

Total Churn (57% of the project completed)	**279%**
Estimated churn for the complete project	**490%**

Table 4.10: Average Churn estimate based on the change control meeting and the Bill of materials.

Using the lifetime churn of the project, the parameter "quality" will be changed to a non-dimensional parameter named "Probability of Change" derived from the lifetime churn. If the lifetime churn has an average of 5 changes per part, it means that the work will have to be completed 6 times; therefore, the probability of change will be the inverse of this or 0.166. This means that 17% of the parts will be designed correctly the first time and will not need rework.

In addition to these parameters, the dynamics of design change control show that there is usually a delay between the need of a change and when it is officially approved, represented by the Parts change approval rate. This delay represents the time to prepare the information to support the change, including the cost and time of the change, as well as the waiting time in change control driven by the limited resource of the project manager. This average delay will be preliminary settled at 4 weeks: 2 weeks for preparing the change and 2 weeks for scheduling and approving it.

In addition, now that the design tasks have changed to the total parts to be designed, the design productivity has also to be updated to the new model with change control. To estimate the average design productivity, based on the company's product development system, the design is frozen several times to allow the integration activities to review the vehicle compatibility. These design check points cycles are longer in the conceptual development phase, and they become shorter as the design progresses. Based on the product development system, the average design cycle and these checkpoints happen on average every three months. Based on the proposed average design cycle, the proposed productivity will be derived from equation 4.11 below using the complete project information (case 4). The complete causal loop model is reproduced in figure 4.24.

$$Design\ Productivity\left[\frac{Parts}{People \cdot Week}\right] = \frac{Total\ Parts\ to\ Design\left[Parts\right]}{Average\ Headcount\left[People\right] \cdot Avg\ Design\ Cycle\left[Weeks\right]} \qquad (4.11)$$

$$Design\ Productivity\left[\frac{Parts}{People \cdot Week}\right] = \frac{4337\left[Parts\right]}{163.6\left[People\right] \cdot 12\left[Weeks\right]} = 2.21\left[\frac{Parts}{People \cdot Week}\right]$$

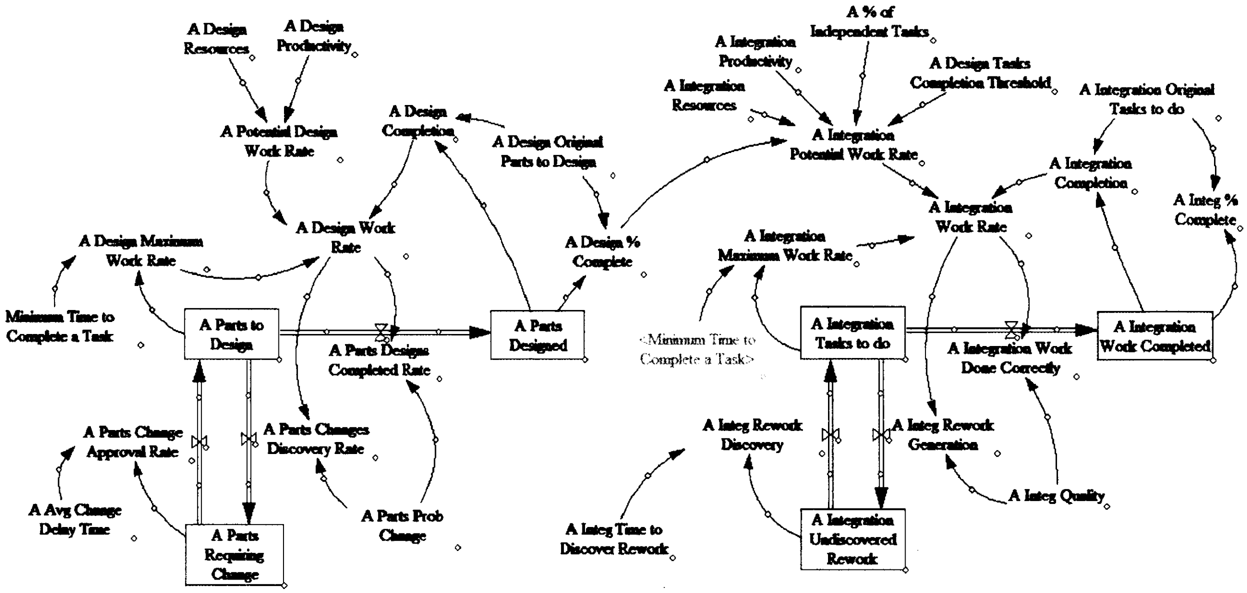


Figure 4.24: Updated Model with Change Control.

Using the updated model, new simulation cases were explored. To overcome the integration rework driven overrun, because the headcount already assumes some rework (per the experience from prior product development projects), the total amount of tasks will be reduced by the same proportion driven by "Integration Quality". The proposed input variables are detailed in table 4.11. The simulation will be run considering two cases, with or without the headcount curve as detailed in figure 4.21 with a 1 month offset between the design and the integration curves. The simulation will monitor the expected completion time in the four selected case studies, for the four selected design phases all with or without the headcount curve. The case without the headcount curve will use the average workforce and the headcount curve will be defined by the peak headcount.

The simulation results are shown in table 4.12. The results for all four case studies are the same regardless if it was with constant or variable headcount (with headcount curve), meaning that all the model tasks and headcount are scaled to the total project scope. The addition of the headcount curve deteriorated the simulated completion time in all cases, and when the complete project is simulated, again the effect of a large project tail is present. In all the phases but the concept development phase, the integration work is finished after the design work, the reason is that 80% threshold is achieved quickly, however, the design churn continues until the design achieves compatibility. The project expected delay is consistent with the observed delay in the actual project, where a 1 month to 2 month delay was reasonable for each of these reviews, however, the only completion time that does not correlate to the actual project is the finalization of the design work in the conceptual development phase, however, this will be corrected in the following section after acknowledging that the total number of parts grows as the project progresses.

Case Study	Design Stage	Required Workforce (Avg / Peak) [People]		Work To Do [Tasks]		Productivity [Tasks/People-Week]		Design Churn [Changes/part]
		Design	Integration	Integration [Tasks]	Design [Parts]	Design	Integ	Design
1	Concept Development	56.2 / 126.8	39.6 / 89.5	760.7	2449	3.63	1.0	0.698
	Detailed Design	78.0 / 126.8	55.0 / 89.5	2817.5	2449	2.62	1.0	1.860
	Testing and Refinement	94.4 / 126.8	66.6 / 89.5	5329.1	2449	2.16	1.0	2.907
	Complete Project	92.4 / 126.8	65.2 / 89.5	8970.1	2449	2.21	1.0	5.000
2	Concept Development	80.0 / 180.7	56.4 / 127.5	1083.7	3489	3.63	1.0	0.698
	Detailed Design	111.1 / 180.7	78.4 / 127.5	4014.0	3489	2.62	1.0	1.860
	Testing and Refinement	134.5 / 180.7	94.9 / 127.5	7592.2	3489	2.16	1.0	2.907
	Complete Project	131.6 / 180.7	92.9 / 127.5	12779.3	3489	2.21	1.0	5.000
3	Concept Development	84.3 / 190.4	59.5 / 134.4	1142.4	3678	3.63	1.0	0.698
	Detailed Design	117.1 / 190.4	82.6 / 134.4	4231.4	3678	2.62	1.0	1.860
	Testing and Refinement	141.8 / 190.4	100.0 / 134.4	8003.5	3678	2.16	1.0	2.907
	Complete Project	138.8 / 190.4	97.9 / 134.4	13471.6	3678	2.21	1.0	5.000
4	Concept Development	99.4 / 224.6	70.2 / 158.4	1347.1	4337	3.63	1.0	0.698
	Detailed Design	138.1 / 224.6	97.4 / 158.4	4989.6	4337	2.62	1.0	1.860
	Testing and Refinement	167.2 / 224.6	118.0 / 158.4	9437.5	4337	2.16	1.0	2.907
	Complete Project	163.6 / 224.6	115.4 / 158.4	15885.3	4337	2.21	1.0	5.000

Table 4.11: Model with design churn change control simulation model variable inputs

			Simulation Results			
Case Study	Design Stage	Planned Completion Time [Week]	Constant Headcount		With Headcount Curve	
			Design Time [Week]	Integration Time [Week]	Design Time [Week]	Integration Time [Week]
1	Concept Development	24.0	38.43	29.62	42	33.81
	Detailed Design	64.0	69.25	70.68	79.19	73.69
	Testing and Refinement	100.0	96.5	108.56	111.88	111.12
	Complete Project	172.0	150.06	184.81	164.19	258.38
2	Concept Development	24.0	38.5	29.62	42	33.81
	Detailed Design	64.0	69.25	70.68	79.19	73.69
	Testing and Refinement	100.0	96.5	108.56	111.88	111.12
	Complete Project	172.0	150.06	184.75	164.19	258.5
3	Concept Development	24.0	38.5	29.62	42	33.81
	Detailed Design	64.0	69.25	70.75	79.19	73.69
	Testing and Refinement	100.0	96.5	108.62	111.88	111.12
	Complete Project	172.0	150.06	184.81	164.19	258.68
4	Concept Development	24.0	38.5	29.56	42	33.88
	Detailed Design	64.0	69.25	70.68	79.19	73.69
	Testing and Refinement	100.0	96.5	108.56	111.88	111.12
	Complete Project	172.0	150.125	184.87	164.19	259.81

Table 4.12: Model simulation results. Model was updated assuming design churn and change control with and without headcount curve.

4.7 Incorporation of Design Scope Increase into the Model

The case study in chapter 3 described how the total number of parts increased throughout the project. In order to overcome the significant schedule slip in the prior simulations for the concept development stage and to more accurately simulate the actual project, a new flow will be included in the project, corresponding to the increase of parts-to-be-designed. Table 4.13 illustrates how the total number of parts steadily increased throughout the project for the wave 1 vehicles. A similar behavior was found also in wave 2, however, wave two data will not be used as a data input for the simulations.

	Breakdown			Week 60	Week 88	Week 100
Unique Parts	Egyptian Dog (ED)			468	562	534
	Greek Dog (GD)			1299	1974	2119
	Mayan Dog (MD)			185	207	236
	Retriever (R)			661	835	881
Used in 2 Derivatives	ED	GD		124	112	106
	ED	MD		45	74	67
	ED	R		125	176	156
	GD	MD		404	210	238
	GD	R		14	30	29
	MD	R		2	3	2
Used in 3 derivatives	ED	GD	MD	446	457	449
	ED	GD	R	133	195	122
	ED	MD	R	24	45	27
	GD	MD	R	26	19	20
Shared for Wave 1	All wave 1 - Dog Products			381	350	291
Total				4337	5249	5277

Table 4.13: Total part numbers increase throughout the project.

Using a linear regression using these three data points, the total number of parts were be included into the model assuming that the parts count increases linearly. A different equation will be derived for each of the four case studies, as the part count increase was different for each project. The slope and y intercept of each equation is shown in table 4.14 below. These values were included into the model, assuming that the parts increase will stop at the planned completion date to avoid the effect of a long tail. A new set of simulations will be investigated to include the effect of parts increase into the model.

Case Study	Project A	Project B	Y Intercept (Initial Value) [Parts]	Slope [Parts/Week]
1	Egyptian Dog	Retriever	2050	7.42
2	Egyptian Dog	Greek Dog	2439	18.29
3	Egyptian and Greek Dog	Mayan Dog	2553	19.49
4	Egyptian dog and Retriever	Mayan and Greek Dog	2879	25.11

Table 4.14: Parts increase equations for each case study.

The simulation was run using the same input values as the prior case (table 4.12), considering the headcount curve only. The results of adding the parts increase into the model did not change the behavior significantly from the prior model as shown in table 4.15. Only the concept development phase reduced the completion time, as expected; however, all the other phases increased their completion time significantly. On the other hand the different projects started to have some project completion time variation among them in their different check points, driven by the correlated data from the actual case study information.

Case Study	Design Stage	Planned Completion Time [Week]	Completion time with parts increase		Completion time without parts increase	
			Design Time [Week]	Integration Time [Week]	Design Time [Week]	Integration Time [Week]
1	Concept Development	24.0	**41**	33.43	42	33.81
	Detailed Design	64.0	86.19	73.31	79.19	73.69
	Testing and Refinement	100.0	131.68	110.81	111.88	111.12
	Complete Project	172.0	179.81	259.125	164.19	258.38
2	Concept Development	24.0	**40.25**	33.06	42	33.81
	Detailed Design	64.0	89.5	72.94	79.19	73.69
	Testing and Refinement	100.0	137.68	110.43	111.88	111.12
	Complete Project	172.0	180.5	256.12	164.19	258.5
3	Concept Development	24.0	**40.25**	33.06	42	33.81
	Detailed Design	64.0	89.56	24.57	79.19	73.69
	Testing and Refinement	100.0	137.81	110.37	111.88	111.12
	Complete Project	172.0	180.62	255.98	164.19	258.68
4	Concept Development	24.0	**40.12**	33	42	33.88
	Detailed Design	64.0	90.25	72.88	79.19	73.69
	Testing and Refinement	100.0	138.87	110.31	111.88	111.12
	Complete Project	172.0	181.56	256.5	164.19	259.81

Table 4.15: Simulation results comparing the model with and without the parts increase effect.

4.8 Incorporation of Average Design Issues Discovery Rate and Apparent Progress.

The model has evolved to capture relevant dynamics in product development, driven by design churn; however, the model has not yet been able to capture the perceived progress of the project instead of the real progress in the various design phases. The model has used an expected average part churn as the main input for the different phases. Yet, while the model has assumed higher design productivity during the concept development phase, it still had a significant simulated schedule overrun during this phase.

During the conceptual development phase, design churn is expected to be lower (as shown in the change control graph (figures 3.6 to 3.8) and eventually ramp up, similarly to the headcount curve. The reason for the slow start is not that the design is issues free, but that these issues have not yet been discovered. These issues are discovered later in the process after the designs are validated either with virtual tools or with engineering prototypes and testing. Even with those uncertainties, the project is able to complete the various development phases because the progress is consistent with the expected delivery.

In order to capture those dynamics in the model, the parts stocks will be updated with a new stock named "undiscovered changes", which will be consumed using a design changes discovery rate which will be modeled as a first order delay using an average delay discovery time. The delay discovery time will be proportional to the time delay to the next virtual assessment and the next engineering prototype. The delay discovery time is considering the following information

- Delay between design checkpoints: average of 14.5 weeks
- Time from project start to the first (underbody) prototype: 74 weeks
- Time from project start to the complete physical prototype: 124 weeks

The design checkpoints are the events where the integration activities request a design freeze to perform their tasks. This is consistent with the assumed average design cycle of 3 months included into the model for the design productivity. At these design checkpoints, the integration activities assess the design using engineering analytical models or computational tools to verify the design, before the prototype is build. The average delay time was calculated as a weighted average for the average design checkpoints schedule and the delay to the engineering prototypes. The weighted value was selected to be proportional to the confidence level of the analytical and computational verification tools, as they are usually unable to predict all the design failure modes with accuracy. This was assumed to be an 80% confidence, or in other words, 80% of the design changes can be verified accurately with these analytical tools. Equation 4.12 below represented the average change discovery delay:

$$\text{Avg Change Discovery Delay} = \frac{\text{Computational Models Accuracy}\ \text{Delay between design Checkpoints} + \left(\text{Computational Models Accuracy} - 1\right)\ \text{Physical Prototype Verification Delay}}{2} \tag{4.12}$$

As the project progresses, the changes discovery delay creates an apparent progress in the project where the parts with undiscovered changes are thought to be complete. This erroneous perception is common and will be included into the model as the input to the integration activities. These model updates are shown in figure 4.25.

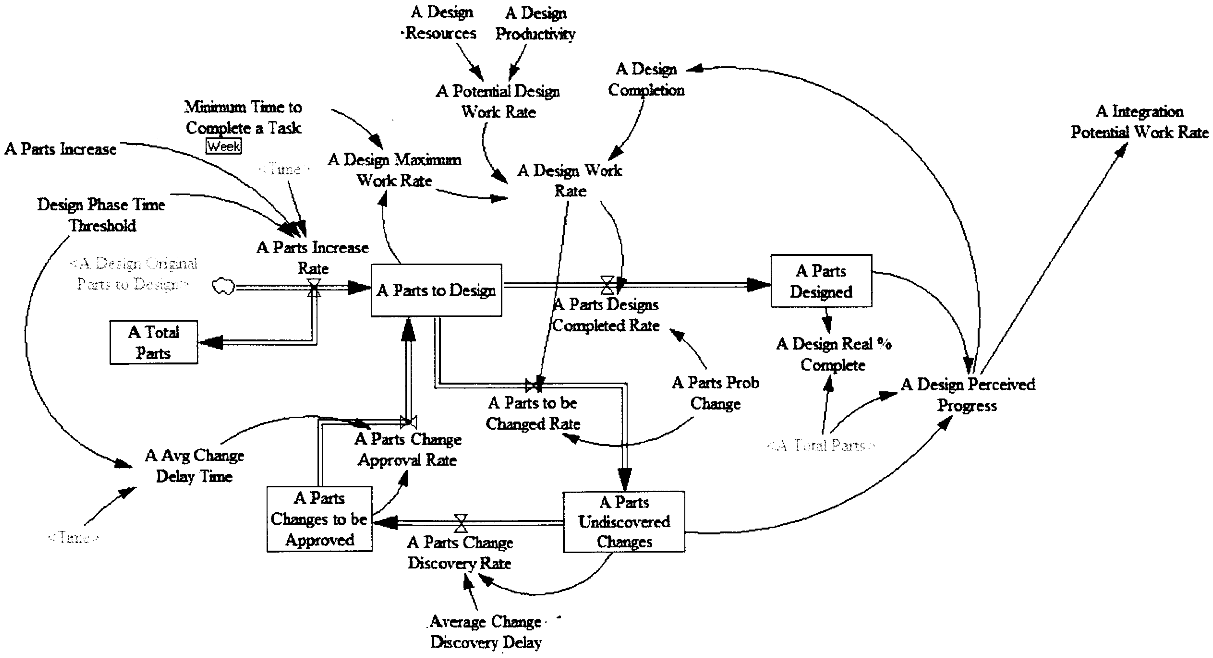


Figure 4.25: Updated model with average change discovery delay and perceived progress.

The new model setup will allow the project to end when the perceived progress reaches 99%, nevertheless, this will happen only when there are no changes to be approved left. In order to achieve this, the normal project behavior is to process these changes faster in order to deplete these changes completely. This reality will be incorporated into the model affecting the change delay time by reducing it towards the end of the phase. The delay time will be reduced linearly from a maximum of 4 weeks to 0.25 weeks 15 days before the scheduled completion date and the rate increase should begin 3 months before the deadline, the change delay behavior is shown in figure 4.26. In addition figure 4.27 shows a simulation run on the behavior of the changes to be approved that reach a maximum quickly and then it is gradually reduced with and without incorporating the increased changes rate towards the end of the phase as found in the case study, the quicker ramp down represents the management effect to prioritize all those changes towards the end of each phase.

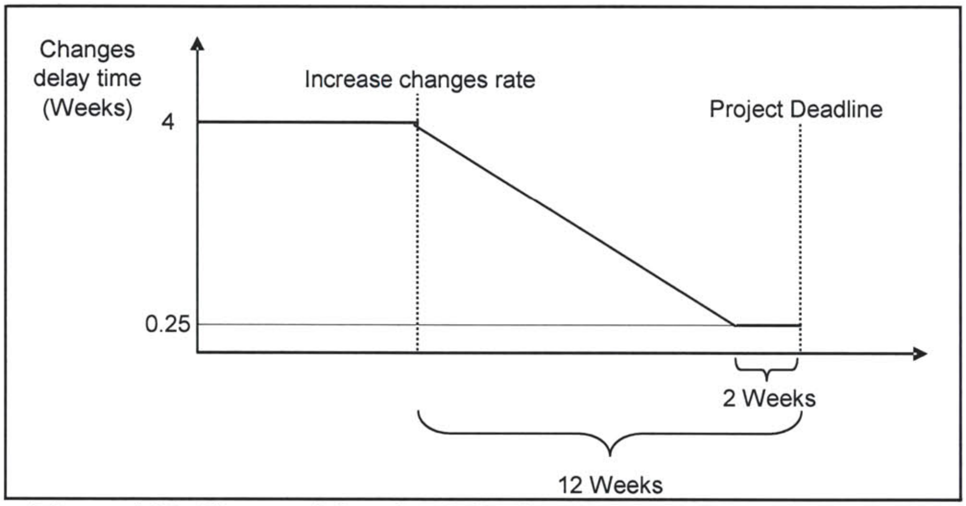


Figure 4.26: Changes delay time behavior to be included into the model

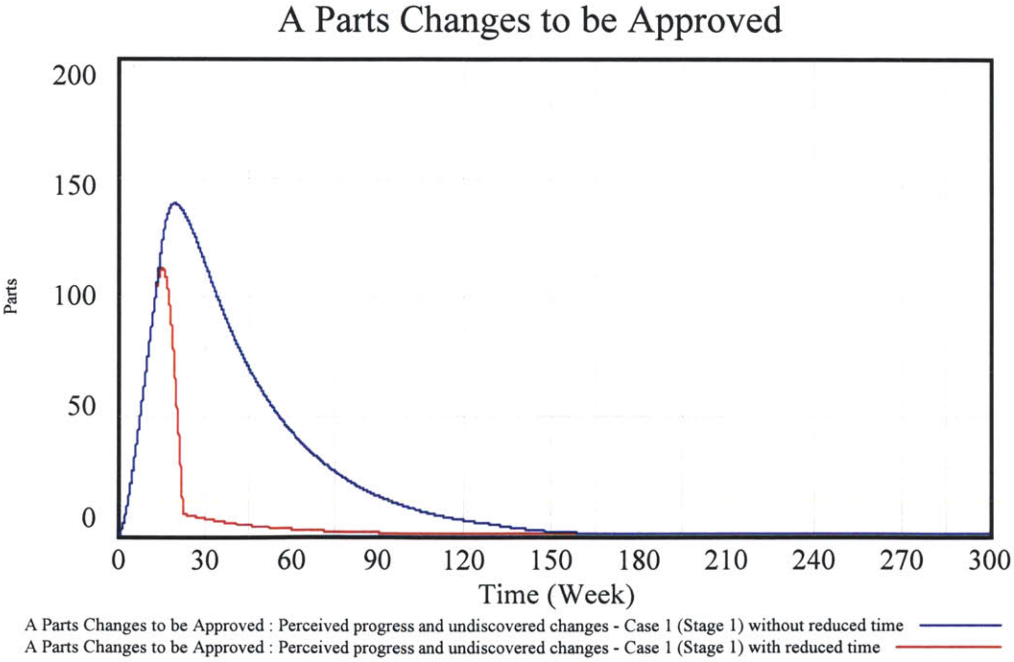


Figure 4.27: Behavior of the pending changes to be approved.

An important feature that was missed on the prior version of the model is the fact that when more parts are added to the project, not only the design activities workload increase, but also the integration workstreams will require more time to complete. In order to capture this behavior, a general relationship of 3.66 integration tasks per part was found instead of a fixed number of integration tasks to be simulated. The updated portion of the model considering the scope increase in integration tasks due to additional parts is shown in figure 4.28:

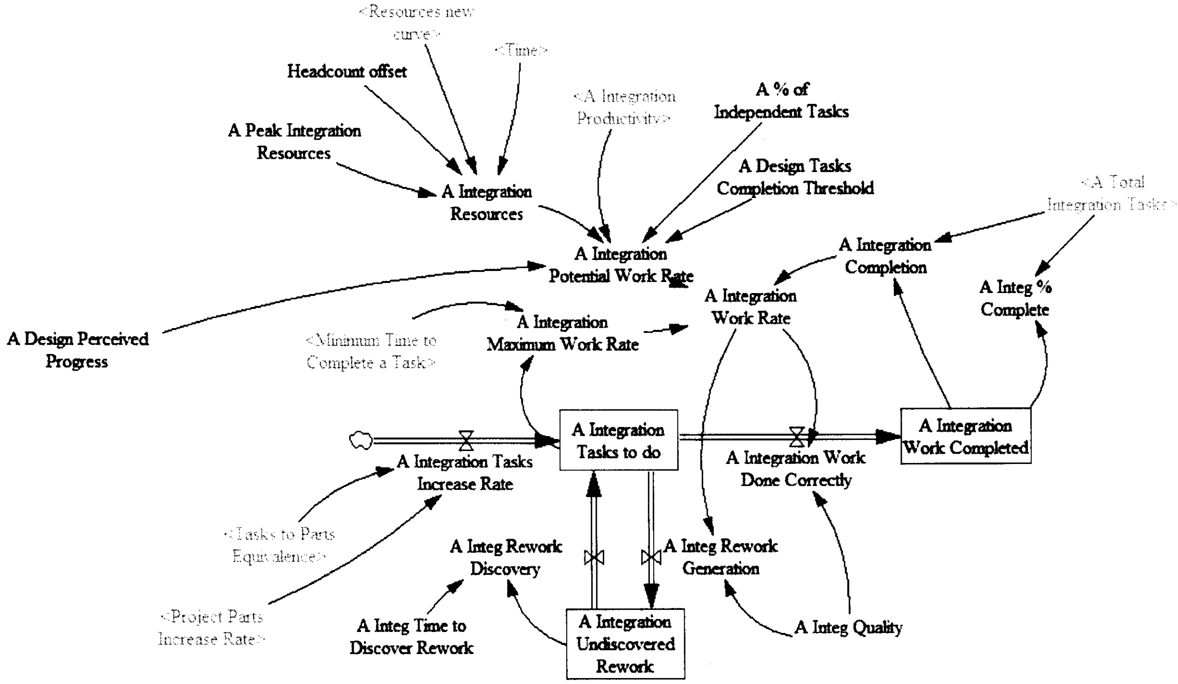


Figure 4.28: Updated integration activities model with scope increase driven by parts increase.

Similar to the design activities, the long tail effects of the simulation as the project approaches the deadline are important. The integration activities will also incorporate an increase in the work rate as the final tasks are being completed. In this case, three months prior to the project deadline the integration productivity will increase gradually to a 20% increase in productivity that will be sustained until the tasks are completed. The productivity increase is shown in the figure 4.29 below.

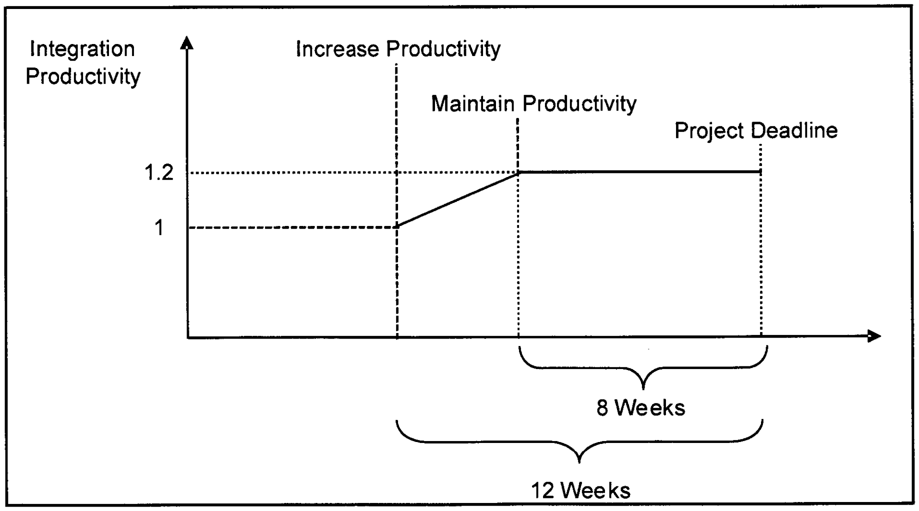


Figure 4.29: Integration productivity increase towards the project deadline.

Finally, the model was also updated using the perceived progress as the design work "switch" instead of the real progress as used in the prior phases, however, the simulation was still expected to end when 99% of the perceived work is completed. This model update should only be used when a partial project is being simulated, however, if the complete project were to be simulated, the real progress should be used instead. This setup creates a simulation apparent error where the perceived progress fluctuates around 99% after the first time the project is completed when the parts completion rate is plotted over time (see figure 4.30). However, to measure the completion date, the project schedule will be recorded as the first time the 99% completion level is achieved. Table 4.16 summarizes the design and integration completion time for the design and integration activities for the three simulated project cut off dates; the complete project scenario will not be simulated from now on as there is no data available from the case study for correlation.

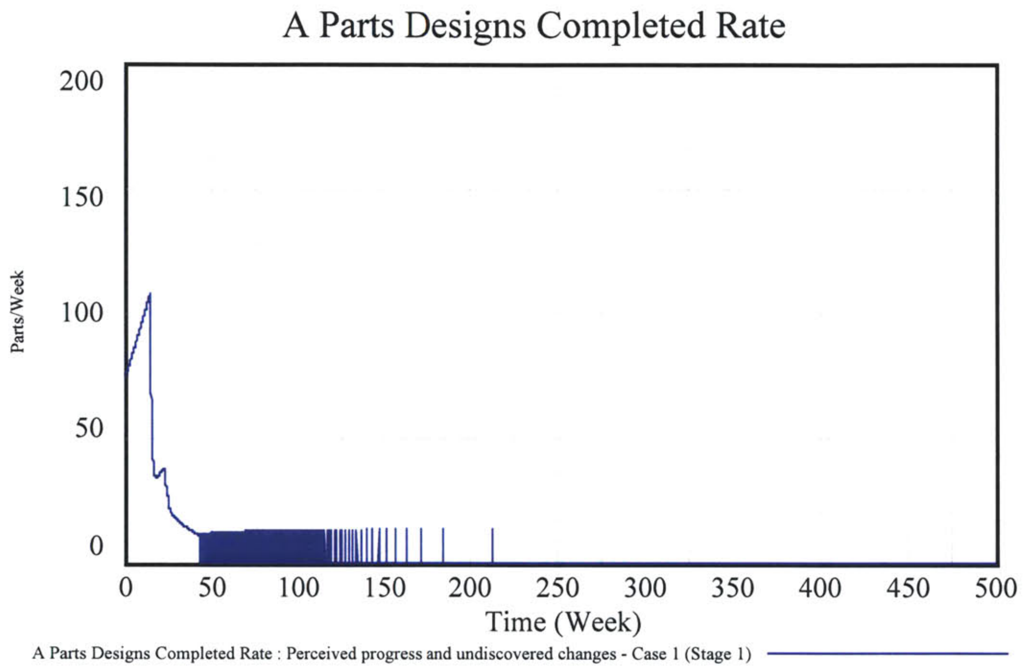


Figure 4.30: Completion rate error after 99% of the apparent project is achieved.

			Simulation Results					
			Completion time with perceived progress (98%)		Completion time with perceived progress (99%)		Completion time without perceived progress	
Case Study	Design Stage	Planned Completion Time [Week]	Design Time [Week]	Integration Time [Week]	Design Time [Week]	Integration Time [Week]	Design Time [Week]	Integration Time [Week]
1	Concept Development	24.0	25.125	26.56	42	27.8	41	28.7
	Detailed Design	64.0	66.56	68.75	98.63	69.9	86.19	72.1
	Testing and Refinement	100.0	105.56	112.37	150.62	113.4	131.68	117.8
2	Concept Development	24.0	25.31	24.75	41.56	25.9	40.25	26.8
	Detailed Design	64.0	68.43	68.69	100.88	69.9	89.5	72.0
	Testing and Refinement	100.0	110.875	117.56	155.68	118.6	137.68	122.9
3	Concept Development	24.0	25.3125	24.62	41.56	25.8	40.25	26.7
	Detailed Design	64.0	68.5	68.62	100.87	69.8	89.56	23.5
	Testing and Refinement	100.0	111.06	117.62	155.88	118.7	137.81	122.9
4	Concept Development	24.0	25.375	24.25	41.50	25.4	40.12	26.3
	Detailed Design	64.0	69.06	68.69	101.56	69.9	90.25	71.9
	Testing and Refinement	100.0	112.25	118.87	157	119.9	138.87	124.1

Table 4.16: Simulation results including the perceived progress and increase in the changes rate.

The first conclusion about the results in table 4.16 is that the new project updates did not change the behavior of the project significantly as expected, however, these updates are further improvements to the model that capture other relevant inputs to project behavior such as the perceived progress and the increased rate of changes towards the end of the milestone. The behavior of the projects is still similar between all the case studies, however, the parts increase rate and initial value per project creates some differences in the completion time among the projects. The detailed design phase (per the integration time completion) is now completed with minimal project slip (compared to no slip in the original project). The detailed design and the testing and refinement phases adequately represent the project behavior as the actual schedule slipped around 2 months for the detailed design phase and around 4 months for the testing and refinement phase.

Another significant observation in the table above is that the project completion threshold, either 98% or 99% has a non-neglibile impact (over 1 week) in the project completion time. The effect to the design completion time is quite significant, and if the project completion threshold is decreased to 98%, the simulation behaves closer to the actual project behavior and the gap between the completion of the design and integration activities is more accurate. The project completion threshold will be updated to 98% per this observation to cancel the effects of the project completion long tail and asymptotic behavior for the final details of the project. This long tail behavior can be appreciated in figure 4.30, where the project completion rate slows significantly and then increases again towards the planned completion date, when the late changes are prioritized.

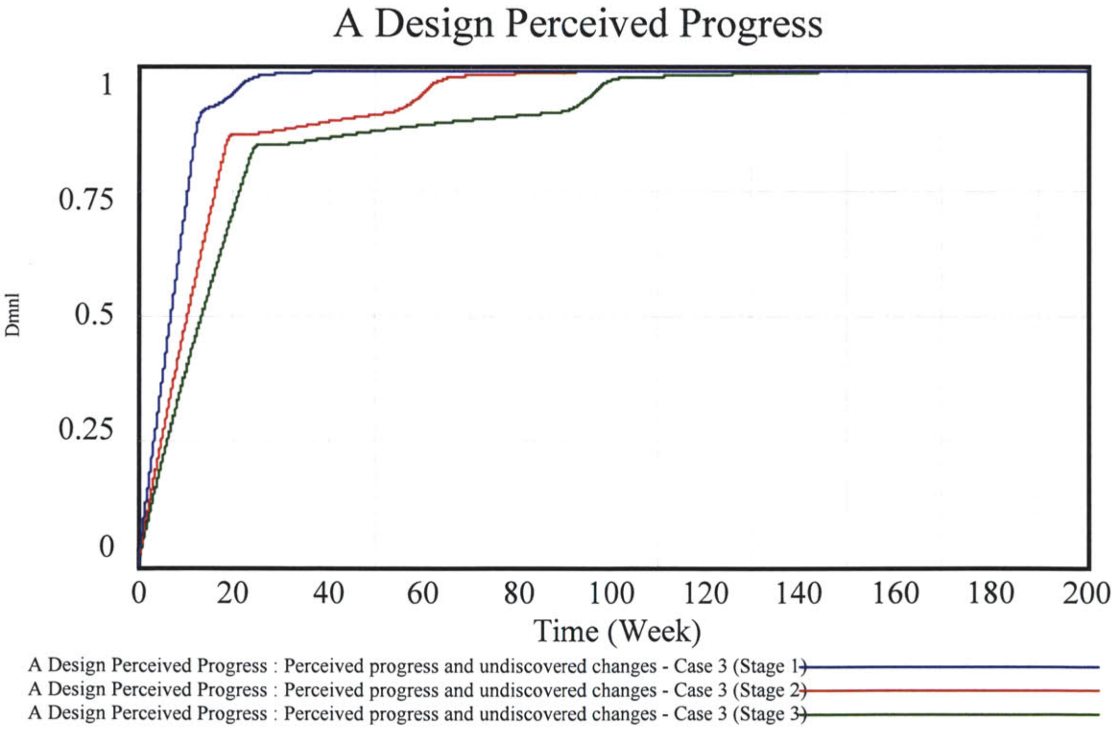


Figure 4.30: Behavior of the project perceived progress of the projects.

4.9 Incorporation of Product Commonality into the Model

To incorporate the effects of product commonality into the simulation, the model was significantly updated from the prior versions. First, the rework cycle for the design activities rework cycle was repeated three times, one for each of the three part classifications available: unique parts for project A (lead project), common parts for project A and project B, and unique parts for project B. Second, the integration activities rework cycle was repeated twice, one for project A and one for project B. Given the significantly increased number of variables, each of these updates will be explained separately. Figure 4.31 illustrates the different "modules" of the simulation to illustrate the overall model structure.

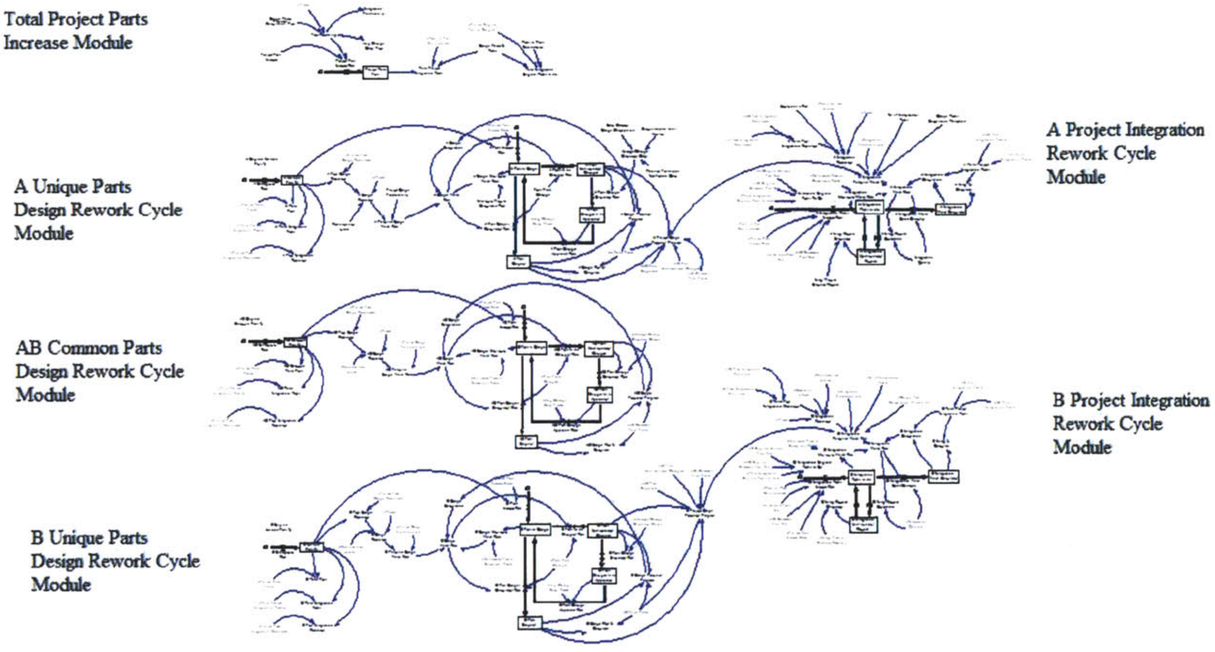


Figure 4.31: Updated model and its "modules" for each of the parts classifications.

Product commonality was simulated as a stock variable, representing a state or a point in time assessment which is affected by the divergence rate. The fraction of common and unique parts was used to scale the model, which already considers the total amount of parts, but not the classification for each of them. As seen in figure 4.29, which represents the left portion of each of the design rework cycle modules, the % of common or unique parts was the relevant variable to scale each of the different inputs to the design and the integration rework cycle as it was used to calculate the total number of parts, integration tasks, integration and design resources, and the parts increase rate for each of the three different parts classifications.

The total number of project parts was also included as a different "module" in the model as illustrated in figure 4.33. This variable was the defined variable to scale most of the project simulation given the available information in the case study; it acted as the main input to measure the progress of the design activities, and also was a relevant input to estimate the integration tasks. In addition, the total number of parts was used indirectly to estimate the number of people; however, the model has assumed a fixed number of resources and a fix resources curve. The figure also illustrates some other relevant inputs to the model like the scheduled completion time and the fraction of project tasks required for each of the simulated phases.

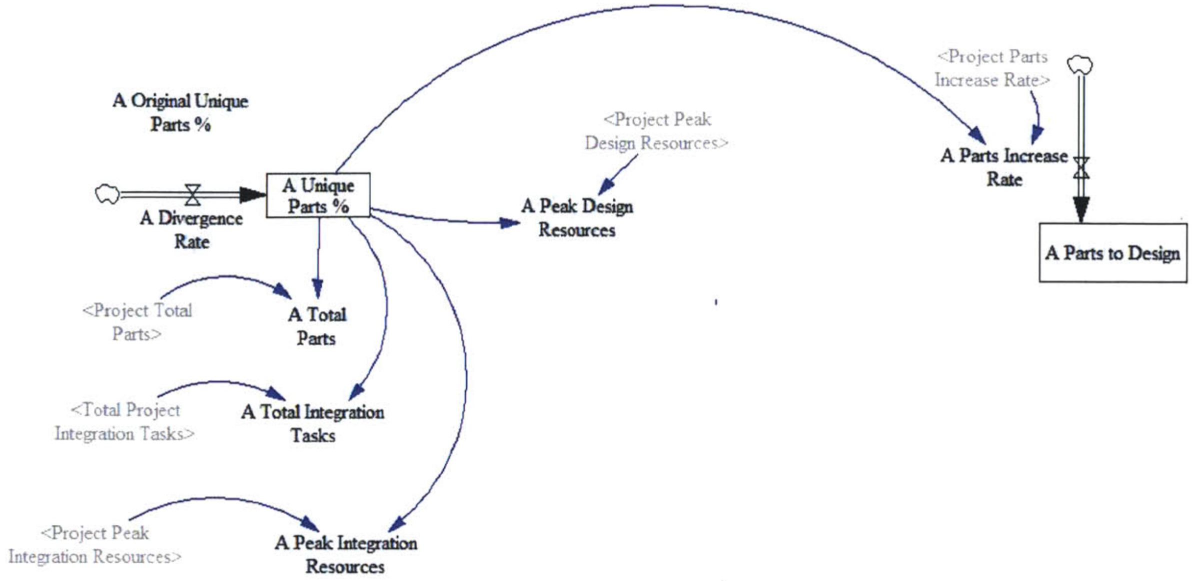


Figure 4.32: Commonality as a stock variable and the additional variables impacted by commonality.

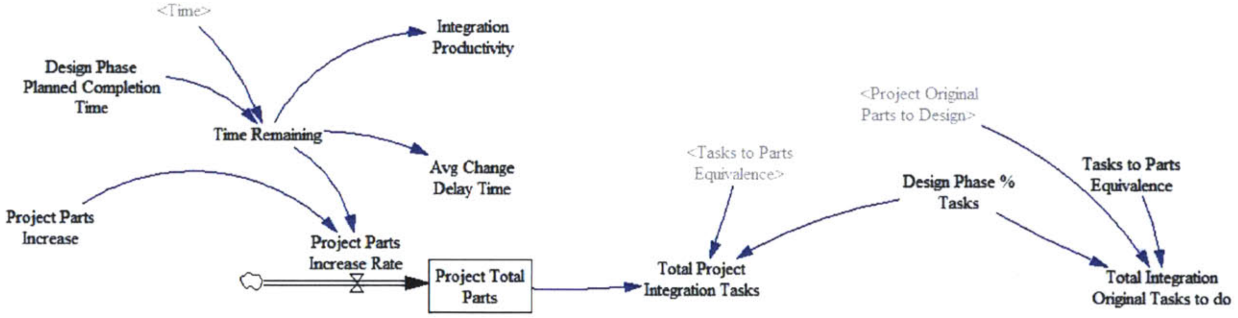


Figure 4.33: Project total parts module.

Each of the design rework cycle modules were modeled with an identical structure as the one shown in figure 4.34. The stocks and flows are the same as the ones explained in the prior section, however, each of the relevant inputs such as the number of parts, the different rates and number of people was scaled based on the % of each of the parts classification. The iterations are primarily controlled by the perceived progress, and all the other variables impacting the rework cycle like the average delays, the productivity, and the parts probability of a change were kept the same for the entire project. The input to the integration rework cycle was the design perceived progress, calculated as the sum of the common and unique parts perceived progress.

Finally, the integration rework cycle module was very similar to the prior model, however, the integration activities will consider the progress from both the unique and the common parts for each project, and all the variables were scaled to the sum of the % of common and unique parts for each of the projects. The integration "quality", average delays, % of independent tasks and the design completion threshold was kept the same for both integration rework cycles. The integration rework cycle module is reproduced in figure 4.35.

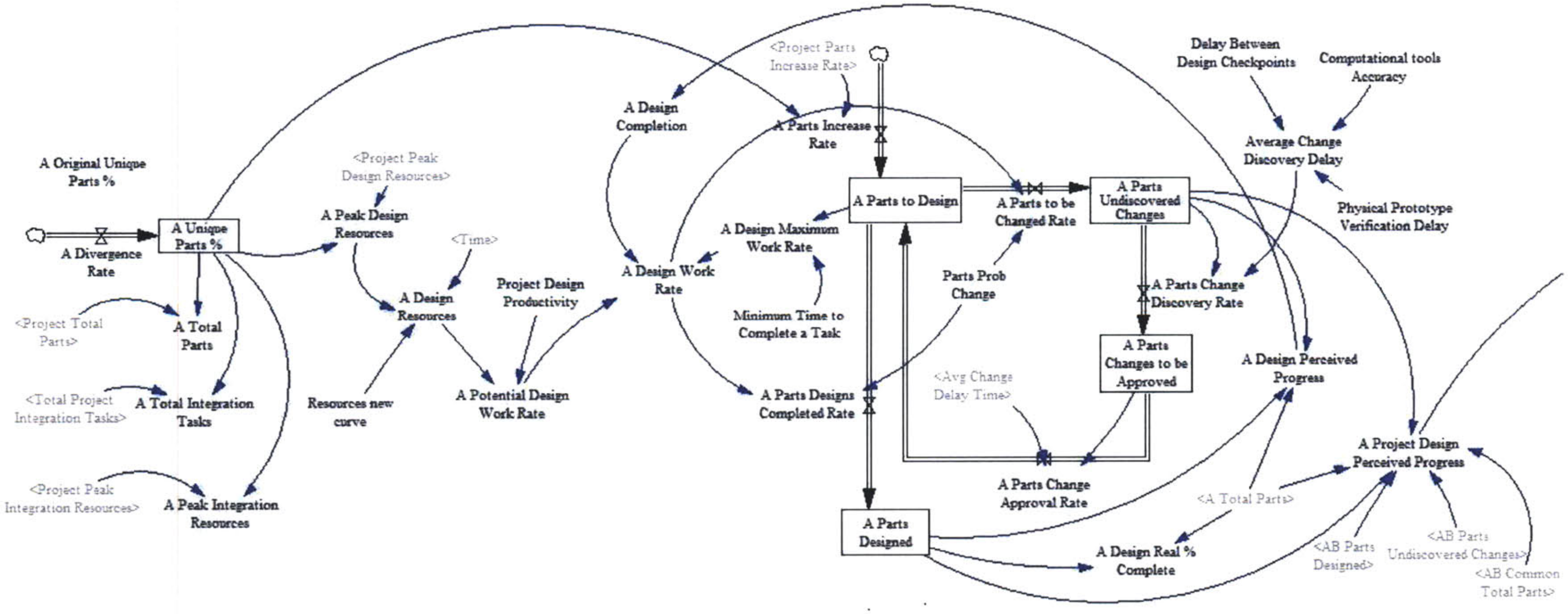


Figure 4.34: Design Rework Cycle Module Structure

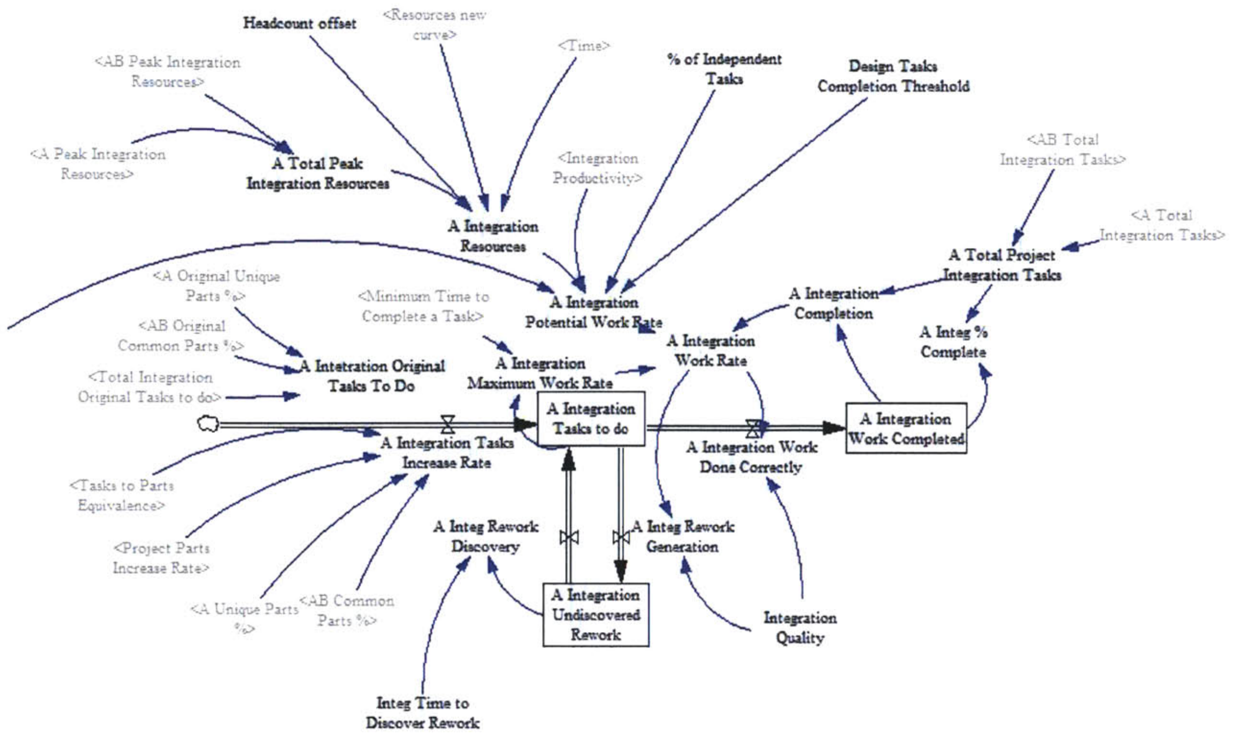


Figure 4.35: Integration Rework Cycle Structure.

Using the new model with common and unique parts for both projects, a new set of simulations were performed. To simulate each of the different projects, in addition to the total number of parts and the required design and integration resources, the model requires inputting the commonality percentages. In this case, while the divergence rate was included in the model, it was set to zero to validate all these model updates, and will be later "turned on" in the model. To simulate the different design phases, the model was updated with the required time for each phase and the percentage of total tasks to be achieved by that phase. Table 4.17 shows all the different input data to the simulations and table 4.18 reproduces the results of these simulations.

Case Study	Design Stage	Planned Completion Time [Week]	% of Total Project [%]	Original Commonality [%] of Total Wave 1 Projects			Required Workforce (Avg / Peak) [People]		Parts Scope Increase		Design Work Rate [Parts/week]	Design Churn Rate - Probability of Part change
				Unique Parts A	Common Parts	Unique Parts B	Design	Integration	Part Increase [Parts/wk]	Design [Parts]		
1	Concept Development	24.0	8.48%	46.19%	33.78%	20.04%	126.8	89.5	7.42	2050	3.63	0.589
	Detailed Design	64.0	31.41%						7.42	2050	2.62	0.350
	Testing and Refinement	100.0	59.41%						7.42	2050	2.16	0.256
2	Concept Development	24.0	8.48%	18.87%	42.58%	38.55%	180.7	127.5	18.29	2439	3.63	0.589
	Detailed Design	64.0	31.41%						18.29	2439	2.62	0.350
	Testing and Refinement	100.0	59.41%						18.29	2439	2.16	0.256
3	Concept Development	24.0	8.48%	41.96%	53.34%	4.70%	190.5	134.4	19.49	2553	3.63	0.589
	Detailed Design	64.0	31.41%						19.49	2553	2.62	0.350
	Testing and Refinement	100.0	59.41%						19.49	2553	2.16	0.256
4	Concept Development	24.0	8.48%	27.54%	36.97%	35.49%	224.6	158.4	25.11	2879	3.63	0.589
	Detailed Design	64.0	31.41%						25.11	2879	2.62	0.350
	Testing and Refinement	100.0	59.41%						25.11	2879	2.16	0.256

Table 4.17: Updated model with commonality: Simulation input data

Case Study	Design Stage	Planned Completion Time [Week]	Simulation Results				
			A Design Time w Common Parts [Week]	B Design Time w Common Parts [Week]	AB Design Time [Week]	A Integration Time [Week]	B Integration Time [Week]
1	Concept Development	24.0	25.125	25.125	25.125	26.56	26.56
	Detailed Design	64.0	66.56	66.56	66.56	68.75	68.75
	Testing and Refinement	100.0	105.56	105.56	105.56	112.375	112.375
2	Concept Development	24.0	25.31	25.62	25.62	24.75	24.75
	Detailed Design	64.0	68.44	68.43	68.43	68.69	68.69
	Testing and Refinement	100.0	110.87	110.87	110.87	117.56	117.56
3	Concept Development	24.0	25.31	25.68	25.68	24.63	24.63
	Detailed Design	64.0	68.5	68.5	68.5	68.63	68.63
	Testing and Refinement	100.0	111	111	111	117.62	117.62
4	Concept Development	24.0	25.69	25.68	25.68	24.25	24.25
	Detailed Design	64.0	69.06	69.06	69.06	68.69	68.69
	Testing and Refinement	100.0	112.18	112.188	112.188	118.87	118.87

Table 4.18: Updated model with commonality: Simulation results.

As seen in the results table, the updated model behaved exactly as the prior model, consistent with the approach to scale each of the different modules by their respective commonality percentage. As explained in the prior section, these results are consistent with the experienced delay time for each of the phases (using the integration time as a reference). The final step of the model calibration is to include the effect of divergence and lifecycle offsets into the simulation.

4.10 Incorporation Divergence and Lifecycle Offsets into the Model

The part sharing studies from Chapter 3 have been used as the main input to the model above. As seen in the case study, there was a clear trend for all the projects to reduce the number of common parts and to increase the number of unique parts. As commonality has been measured as the fraction of common parts to the total number of parts, divergence was decided to be modeled as the percentage of commonality loss per unit of time. Using the data from table 3.10, four different case studies were derived from this data and a general linear relationship was investigated to model the divergence rate behavior throughout the project.

Case Study	Part Classification	Project		Commonality Assessments			Slope	Y Intercept	R^2
1	**Unique Parts**	Egyptian Dog (ED)		44.2%	42.2%	43.1%	-0.037%	46.2%	0.534
		Retriever (R)		28.7%	31.0%	34.7%	0.139%	20.0%	0.878
	Shared Parts	ED	R	27.1%	26.8%	22.2%	-0.102%	33.8%	0.584
2	**Unique Parts**	Egyptian Dog (ED)		19.0%	20.4%	18.9%	0.006%	18.9%	0.025
		Greek Dog (GD)		50.0%	53.1%	57.9%	0.183%	38.5%	0.887
	Shared Parts	ED	GD	31.1%	26.5%	23.3%	-0.189%	42.6%	0.984
3	**Unique Parts**	Egyptian Dog (ED) & Greek Dog (GD)		58.8%	69.0%	69.7%	0.289%	42.0%	0.943
		Mayan Dog (MD)		5.1%	4.8%	5.4%	0.005%	4.7%	0.084
	Shared Parts	ED & GD	MD	36.1%	26.2%	24.9%	-0.294%	53.3%	0.965
4	**Unique Parts**	ED & Retriever		28.9%	30.0%	29.8%	0.024%	27.5%	0.792
		GD and MD		43.5%	45.6%	49.1%	0.128%	35.5%	0.858
	Shared Parts	ED & Retriever	GD and MD	27.6%	24.5%	21.1%	-0.152%	37.0%	0.936

Table 4.19: Divergence rates for the different case studies.

The system dynamics model requires an assumption about the divergence behavior over time. Unfortunately, the literature has not quantified the behavior of divergence over time, therefore, given the limited available data based on three commonality assessments, a linear model was found to be a good approximation, given the high correlation index for most of the curves that describe either the increase of unique parts or the decrease of common parts. Only the first case study did not achieve a good correlation index for the divergence rate (decrease of common parts), which is the major focus of this model.

To further validate if a linear decrease of commonality over time is a valid assumption for the divergence rate, the case study's change control meeting minutes were analyzed and assessed on how many changes impacted either divergence or convergence. Figure 4.36 illustrates graphically how many changes impacted divergence, how many changes impacted convergence (increasing common parts) and how many changes were not impacted by any of these phenomena. The upper bar chart shows the total number of changes and the lower chart shows the divergence rate, or the fraction of changes that impacted divergence or convergence.

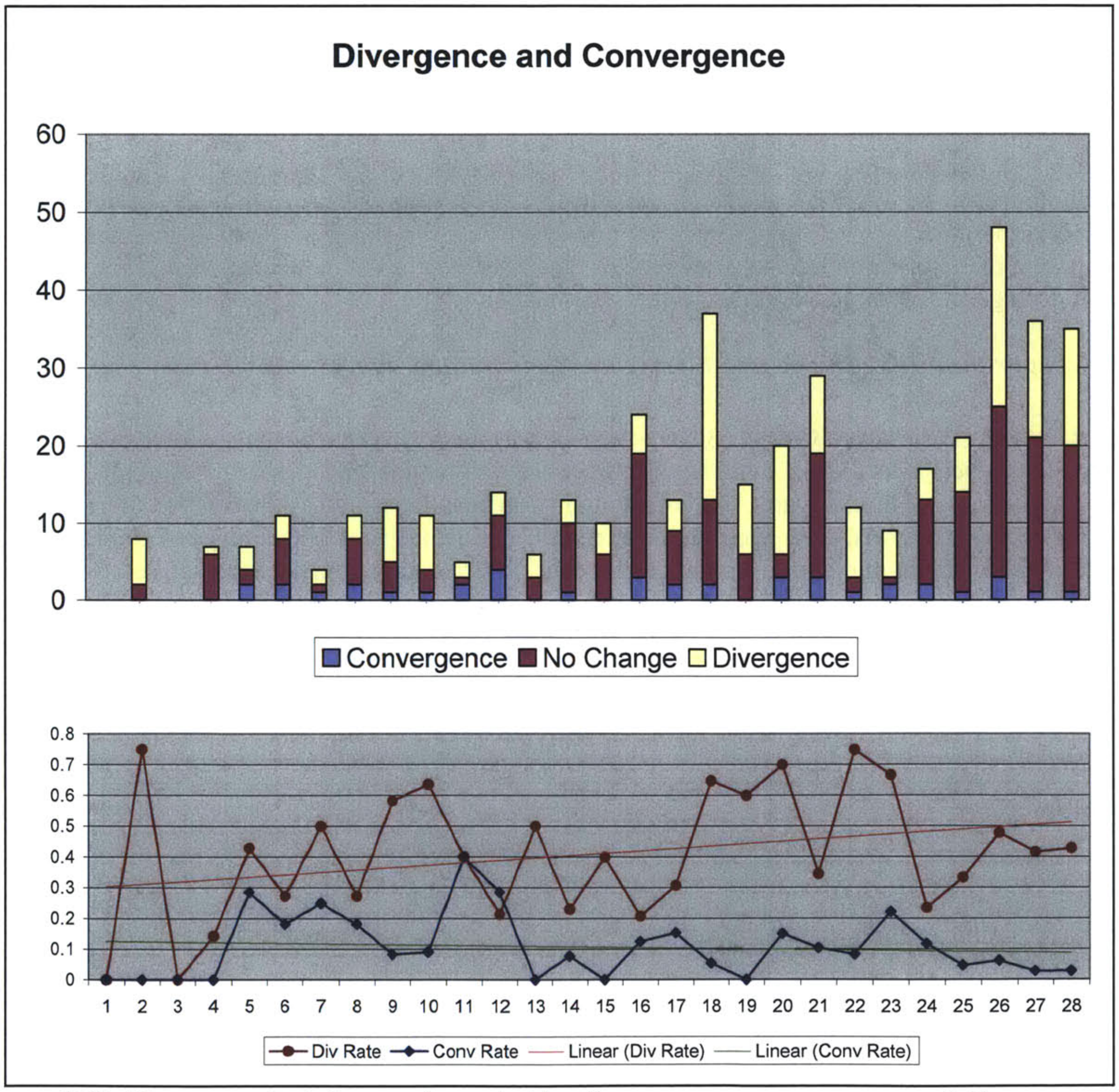


Figure 4.36: Change control changes impacted by divergence or convergence and divergence and convergence rate trend.

The lower chart with the divergence showed that divergence had an erratic behavior throughout the project, however, it grew marginally throughout the project; therefore, a constant divergence rate is a reasonably good assumption for the model. The convergence rate was found to be more constant, and always lower than the divergence rate. The net effect was a higher divergence rate (40% of the changes) compared to the convergence rate (10% of the changes), explaining the commonality loss in the case study.

In addition to including the linear divergence behavior, the effect of the lifecycle offset was also included into the model. The lifecycle offset was 6 months between projects (start of production date) in the case study 2 and 4, where the Greek and the Mesopotamian Dog projects launched their products later than the Egyptian derivatives. This sense of "additional time" was also a significant factor that impacted the development of the product family since most of the attention was captured by the earliest products to be produced.

Even if the development projects were ending with a 6 month offset, the start of the project and all the development phases from the project start to the end of the upperbody detailed design phase were all concurrent. The opposite approach to manage the offset is delaying the project start for the second project; however, developing all the products concurrently is one of the most relevant factors for a quality product and project as concluded by Cusumano and Nobeoka (1998) in their concurrent technology transfer strategy. The risk of this approach is that the feedback and validation from the complete product will be delayed by the same offset. Therefore, the lifecycle offset will be simulated as an incremental change discovery delay at the same magnitude as the lifecycle offset because even the analytic tools feedback will be delayed since all the other unique parts will require more time to be incorporated in the analytic models.

Another required update in the model was standardization of the part increase model and the divergence rate model. A linear model is used to forecast the original product commonality at the beginning of the project, nevertheless, table 4.19 detailed the divergence rates in a percent (exponential) basis instead of the total part count and the parts increase. The model has to be standardized to manage the parts increase in a common way. Table 4.20 below compares the linear increase vs. the exponential increase and their respective rates.

Case Study	Part Classification	Initial Part Count (Forecasted)	Initial Commonality (Forecasted)	Final Part Count (Measured)	Final Commonality (Measured)	Divergence Rate (%/week)	Part increase rate (Parts/Week)	Equivalent Increase rate (%/Week)
1	Unique A	947	46.2%	1156	43.1%	-0.037%		
	Unique B	411	20.0%	932	34.7%	0.139%		
	Common AB	692	33.8%	596	22.2%	-0.102%		
	Total	2050	100.0%	2684	100.0%	0.0%	7.42	0.309%
2	Unique A	460	18.9%	784	18.9%	0.006%		
	Unique B	940	38.5%	2406	57.9%	0.183%		
	Common AB	1039	42.6%	968	23.3%	-0.189%		
	Total	2439	100.0%	4158	100.0%	0.0%	18.29	0.561%
3	Unique A	1071	42.0%	3066	69.7%	0.289%		
	Unique B	120	4.7%	238	5.4%	0.005%		
	Common AB	1362	53.3%	1094	24.9%	-0.294%		
	Total	2553	100.0%	4398	100.0%	0.0%	19.49	0.569%
4	Unique A	793	27.5%	1571	29.8%	0.024%		
	Unique B	1022	35.5%	2593	49.1%	0.128%		
	Common AB	1064	37.0%	1113	21.1%	-0.152%		
	Total	2879	100.0%	5277	100.0%	0.0%	25.11	0.629%

Table 4.20 Parts increase equivalence in an exponential growth perspective and linear perspective.

Figure 4.37 below shows the difference between the exponential and linear parts increase model. As seen in the chart, because the parts increase rate is small, the behavior is almost linear in the proposed model.

In all cases, the linear model yielded a higher parts increase than the exponential growth as it was derived from the final part count measurement and is therefore more accurate to represent the project behavior. In conclusion, the linear model for the total quantity of parts will be used.

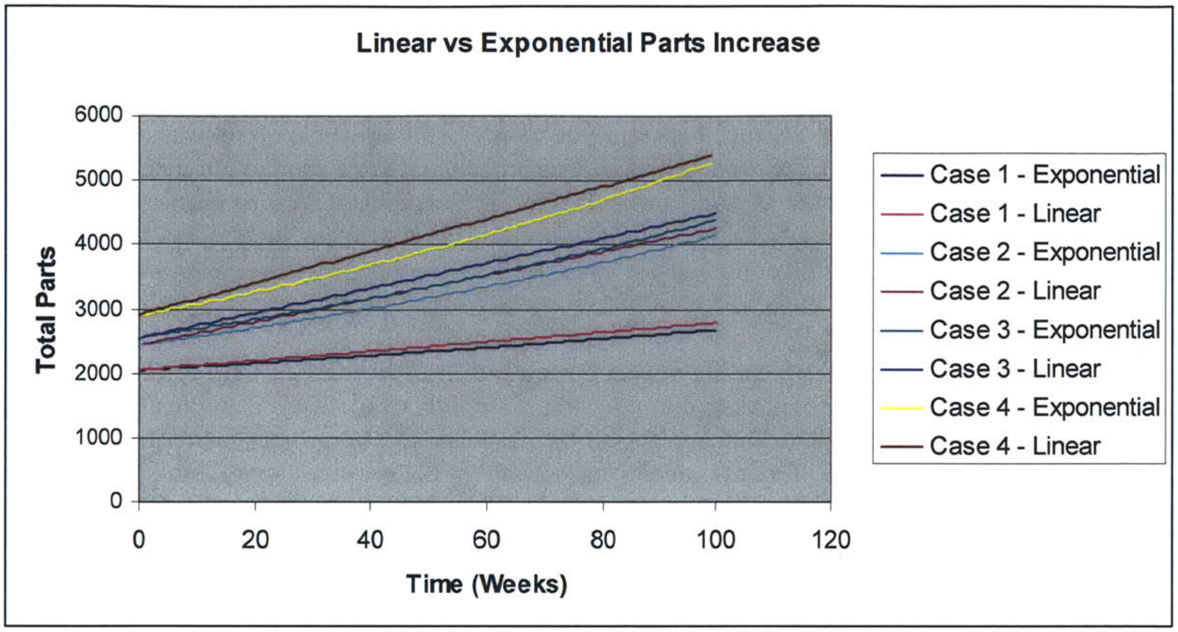


Figure 4.37: Linear vs. exponential parts increase.

Another insight from table 4.20, is that a knock-on effect of divergence is the addition of new unique parts. While the model has already incorporated the effect of the parts increase; many of these parts increases were driven by divergence and it is useful to differentiate the effect from both. When a common part cannot be shared anymore, then it becomes a unique part for one of the products, and another new unique part has to be developed for the other product. This is visually illustrated in figure 4.38 where a design change transformed a common part to a unique part for the first product, and the new part was designed uniquely for the second product, increasing the total part variety and reducing product commonality.

The model incorporates explicitly the parts increase driven by divergence and the parts increase driven by the overall parts increase trend. These updates are shown in figure 4.39. Using the available information of the linear parts increase rate (Total Project Parts Increase Rate), the incremental parts driven by the added scope will be the difference between the total parts and the divergence incremental parts. The divergence Parts Increase Rate will be proportional to the maximum absolute divergence rate for any of the three parts classification that will be proportionally adjusted to the parts to percent rate proportion as shown in the figure 4.39.

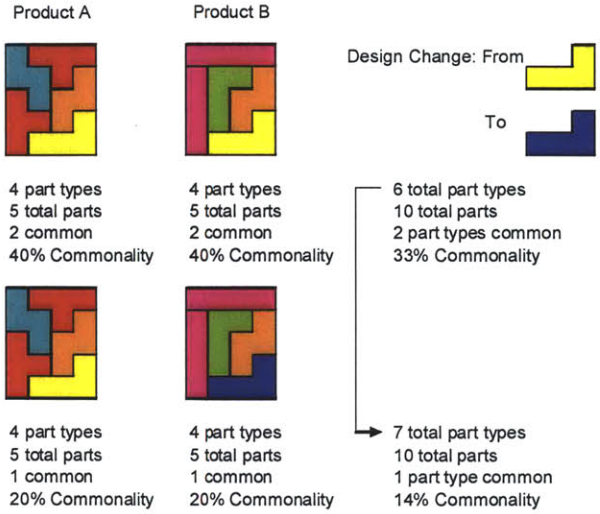


Figure 4.38: Parts increase effect. Adding a new color to the part creates a new part type, increasing the total number of part types and reducing product commonality.

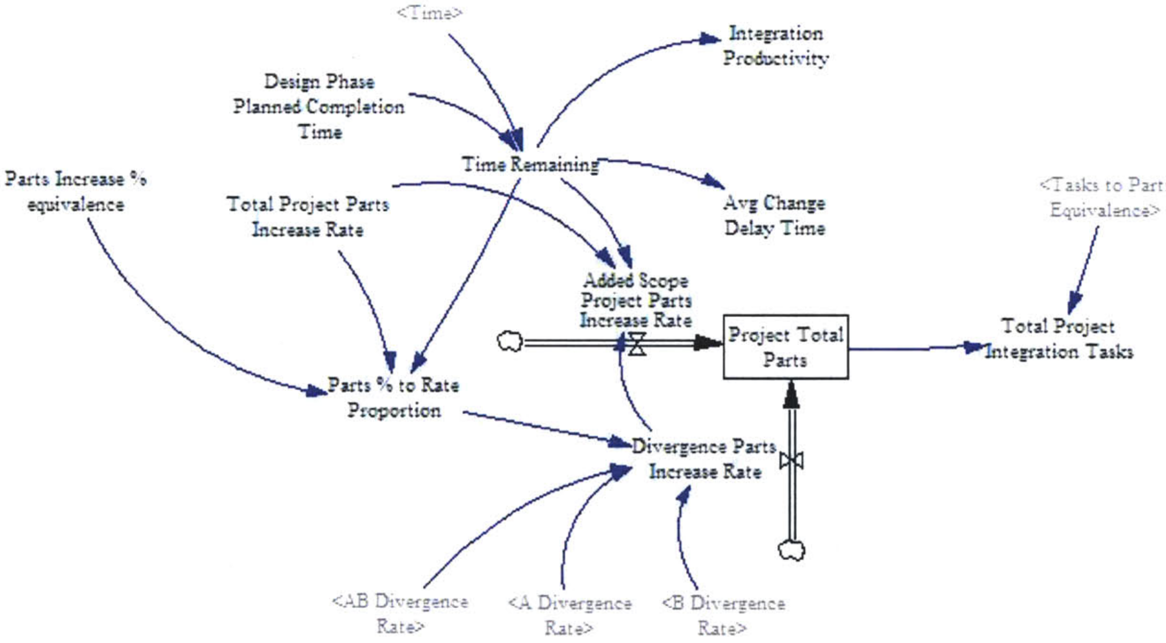


Figure 4.39: Parts increase identification for either scope increase or divergence driven.

The design rework cycle was also updated to capture the parts increase driven by divergence and the parts increase driven by scope increases. In addition, the mix of linear and percent parts increase drove some model inconsistencies on the total number of parts for each of the three parts classifications. In order to solve this inconsistency, the total amount of parts was derived from the sum of the parts in each of the stocks. The updated design rework cycle module is shown in figure 4.40.

After all these model updates, a new set of simulations were performed. The results of these simulations are reproduced in table 4.21. As seen in the table, after the divergence information was included into the model, the project completion date for the different case studies changed. In addition, the simulation results with a 6 month lifecycle offset between the projects (for case studies 2 and 4) was also considered and the results are shown in table 4.22. The lifecycle offsets did not yield a worse project behavior as it would be expected, but actually drove better project completion times. The reason for this is that the measured progress is considering the unidentified changes, which are expected to increase with lifecycle offsets.

To compare these results with the actual project, the completion time for the integration activities will be considered the completion date for the project. In general all the project completion times are consistent with the actual project performance which experienced no delay in the concept development phase, a 4 weeks delay in the detailed design phase, a 10 weeks delay in the testing and validation phase for the lead project and a 16 weeks delay for the follow on project. The comparison of the actual times to the simulation are shown in table 4.23.

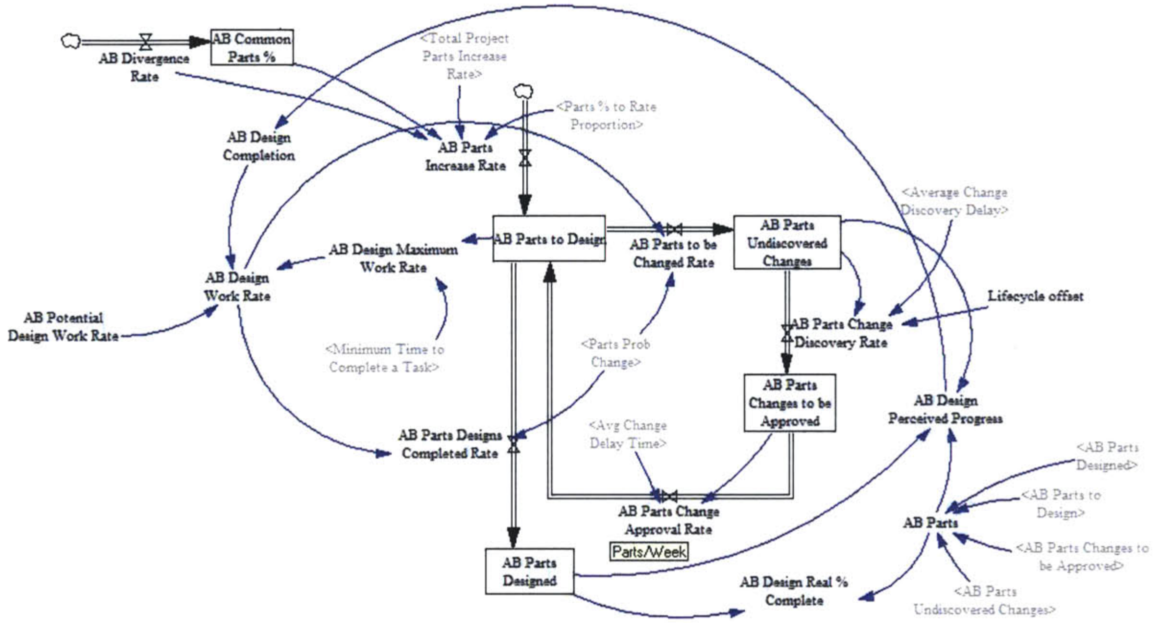


Figure 4.40: Updated model with divergence and lifecycle offsets: Design rework cycle module

Case Study	Design Stage	Planned Completion Time [Week]	Simulation Results without lifecycle offset				
			A Design Time w Common Parts [Week]	B Design Time w Common Parts [Week]	AB Design Time [Week]	A Integration Time [Week]	B Integration Time [Week]
1	Concept Development	24.0	25	23.43	29.75	26.18	26.5
	Detailed Design	64.0	65.5	63.06	71.31	66.69	68.69
	Testing and Refinement	100.0	101.69	104.5	99.125	105.62	112.12
2	Concept Development	24.0	24.678	22.25	29	24.25	24.69
	Detailed Design	64.0	65.1875	63.25	69.5	65.81	68.63
	Testing and Refinement	100.0	103.125	104.75	97	107.68	117.38
3	Concept Development	24.0	25.68	27.62	24.5	24.31	24.43
	Detailed Design	64.0	68.43	80.87	64.62	66.93	68.56
	Testing and Refinement	100.0	111.06	129.37	99.93	112	117.37
4	Concept Development	24.0	25.68	25.94	23.93	23.75	24.25
	Detailed Design	64.0	68.75	73.56	63.12	65.87	68.56
	Testing and Refinement	100.0	111.56	119.43	98.06	109.12	118.313

Table 4.21: Complete model simulation results without lifecycle offsets

Case Study	Design Stage	Planned Completion Time [Week]	Simulation Results (with offset)				
			A Design Time w Common Parts [Week]	B Design Time w Common Parts [Week]	AB Design Time [Week]	A Integration Time [Week]	B Integration Time [Week]
2	Concept Development	24.0	24.875	19.25	27.31	24.18	24.62
	Detailed Design	64.0	65.1875	61.06	68.62	65.69	68.5
	Testing and Refinement	100.0	102.06	101.87	96.25	107.44	117.125
4	Concept Development	24.0	25.18	24.25	21.31	23.81	24.18
	Detailed Design	64.0	69.18	64.56	61.87	65.81	68.44
	Testing and Refinement	100.0	111.68	100.68	97.56	109	118.12

Table 4.22: Complete model simulation results with lifecycle offsets

Case Study	Design Stage	Planned Completion Time [Week]	Actual Project Completion Time [Week]		Case Study	Simulation Results	
			Lead Project	Follow on Projects		A Integration Time [Week]	B Integration Time [Week]
1	Concept Development	24	24	24	1	26.18	26.5
					2	24.25	24.69
					3	24.31	24.43
					4	23.75	24.25
2	Detailed Design	64	68	68	1	66.69	68.69
					2	65.81	68.63
					3	66.93	68.56
					4	65.87	68.56
3	Testing and Refinement	100	110	110	1	105.62	112.12
			110	116	2	107.68	117.38
					3	112	117.37
					4	109.12	118.313

Table 4.23: Model comparison to actual project performance.

An interesting observation on the results above is that usually the follow on project required more time to finish, even if the input data was derived from the divergence estimates and the bill of materials. The actual project behaved in the same way; however, the improved performance of the lead project was driven by the larger attention given to the project.

In conclusion, the current model adequately represents the project behavior compared to the data available and is calibrated accordingly. The model will now be used in the following chapter to develop "what if" scenarios for a hypothetical project to develop general recommendations to improve project performance for product family development where parts commonality is important.

5. System Dynamics Model and Case Study Findings

5.1 Platform Projects Causal Loop Diagram: Conceptual Framework for Tradeoffs

Designing product platforms and achieving common designs requires a profound understanding of the tradeoffs of common designs. Based on the literature review (Meyer & Lehnerd, 1997; Sanderson & Uzumeri, 1995; Cusumano & Nobeoka, 1998) the expected benefits for common designs compared to unique designs include:

- Development and investment cost should be lower for common designs because these costs should be spent for leveraging two or more products;
- Variable cost should be lower for common designs, based on the economies of scale and increased production volumes for two or more products;
- Product quality is expected to be better, because common parts are extensively tested; and,
- Development time is reduced for follow on derivatives, since the part should be designed already.

However, some of the tradeoffs of common designs are:

- Not achieving sufficient product differentiation and therefore product demand can be lower than expected;
- Development time is expected to be equal or longer for common designs, since these designs have to be validated and designed to comply with the requirements of the different vehicles; and,
- If quality issues occur, more products could experience these issues. This was recently experienced by Toyota and its recent product recalls in 2010.

When a development project starts, all these anticipated benefits are quantified and included in the project plan and the project budget. These planning assumptions and expected commonality benefits are used to establish the original project iron triangle that will define the objectives of the product family. Figure 5.1 in next page represents the complex causal loop diagram for platform project management where two projects are represented, project A and project B, where A is the lead derivative and B is the follow on derivative. The following objectives for both projects are established before the project starts, to determine a feasible project iron triangle in terms of cost, scope and schedule, and are identified in pink color in figure 5.1:

- Project schedule;
- Project human resources (budgeted people);
- Overall project Investment (including the development costs: people and prototypes);
- Objective production cost; and,
- Objective vehicle functional attributes.

Numerous studies are performed before the project starts to guarantee the project will be compatible with all the five objectives above; however, there is high uncertainty on these objectives which were based on estimates from surrogate products. In addition, management pressure also expects project over project improvement and additional efficiencies may be assumed on top of the expected product family benefits. Furthermore, the high uncertainty can only be reduced by actually executing the product development process which will inevitably create several changes to the product until it finally reaches its final design. The sum of all these factors produces an incompatible iron triangle from the beginning of the project, which the development team and the project manager attempt to reconcile by further increasing the number of changes to the product in order to meet the five objectives. The dynamics of the added changes are clarified in figure 5.2 which represents only a portion of the complete causal loop diagram that emphasizes project dynamics driven by an incompatible iron triangle that generates the rework cycle.

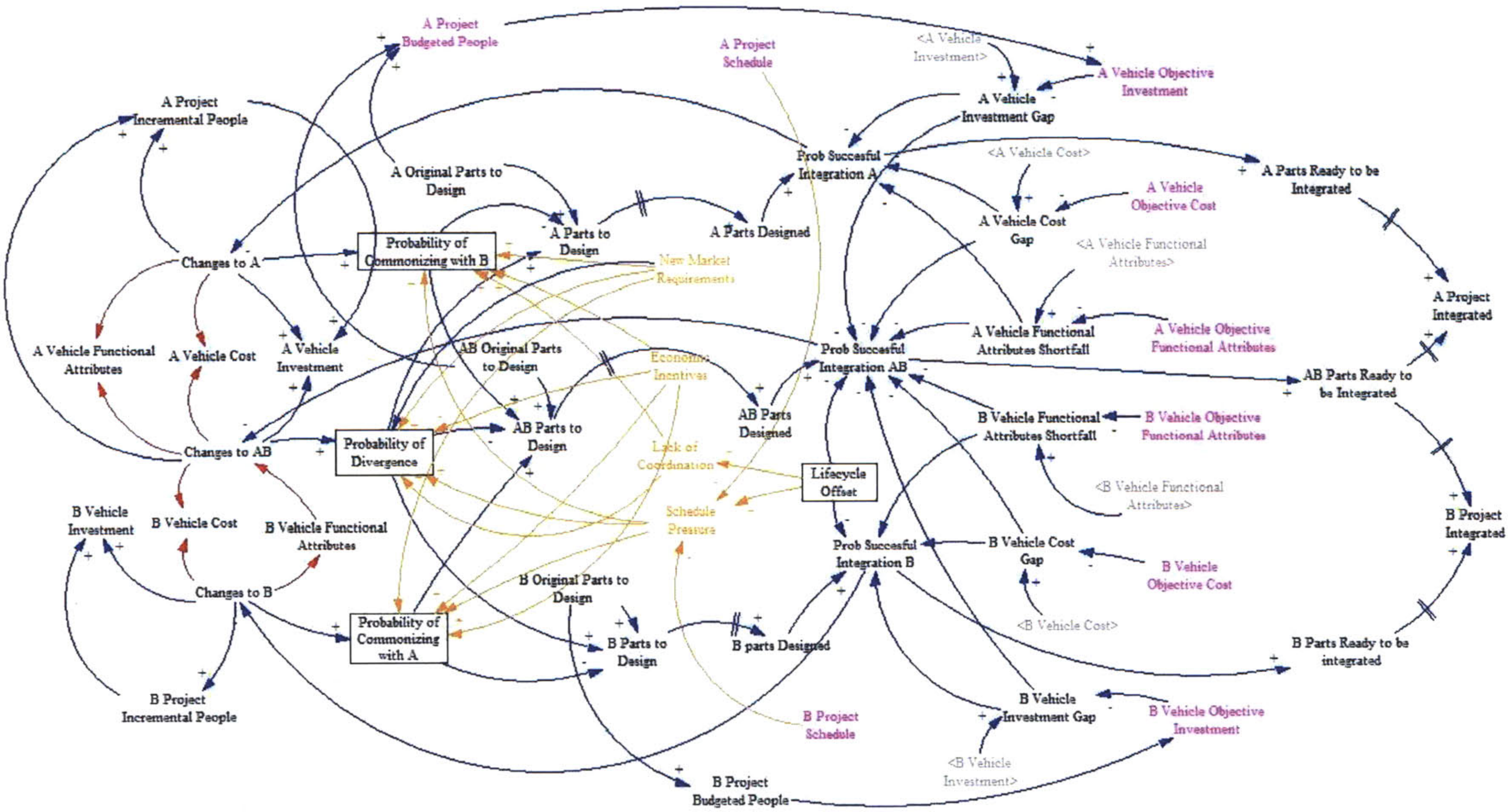


Figure 5.1: Complete Causal Loop Diagram for a two derivatives product family

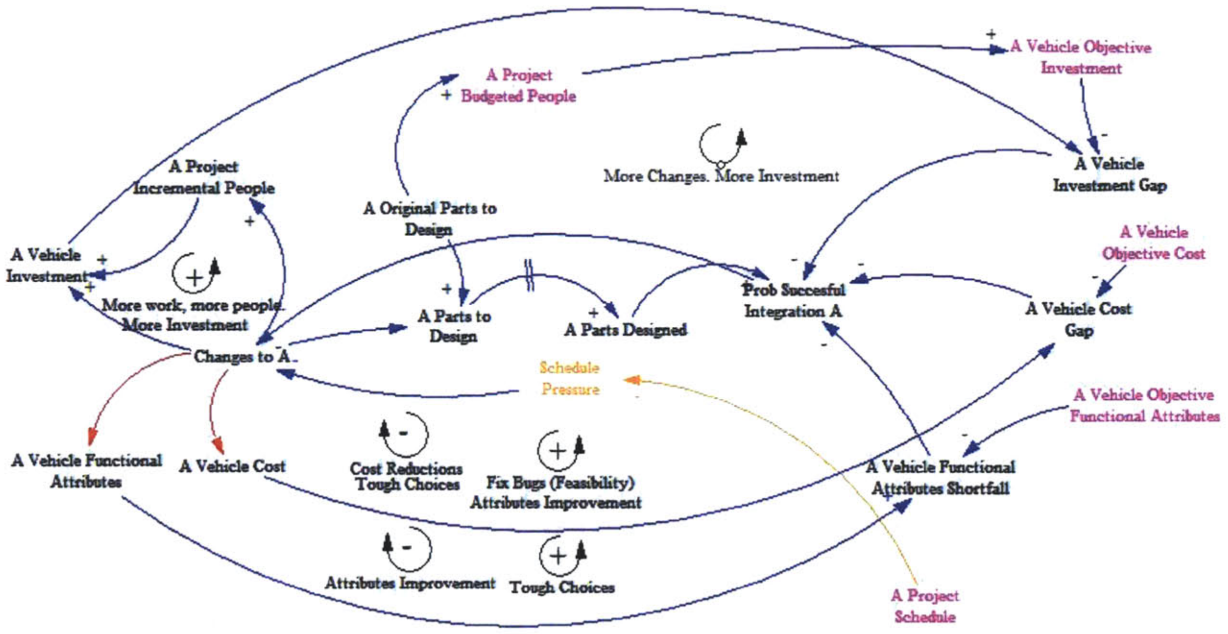


Figure 5.2: Single project causal loop diagram (all parts are unique).

As seen in figure 5.2, in order to maintain a healthy project in terms of schedule, costs (investment costs and product costs) and scope (functional attributes), the project manager must investigate and execute individual changes to the product in order to achieve a compatible project and these changes effectively become the design optimization cycles. The two arrows in the causal loops diagram indicate the uncertain nature of changes, since some times the changes are pursued for lower costs, and sometimes for improved functional attributes; however, it is hard to find changes that improve both functional attributes and costs at the same time. The six feedback loops above are:

- Cost reductions: This feedback loop allows the project manager to change the product to achieve lower production costs. The usual tradeoff is a higher investment and decreased functional attributes.
- Fix bugs (feasibility): This feedback loop is present after design issues are found and they have to be fixed to sell the product. This usually requires an increase in the product cost.
- Attributes improvement: It appears twice in the causal loop diagram, to indicate the tradeoff to balance (improve) the functional attributes, however, it usually increases the cost of the product.
- Tough choices: This also appears twice, as it indicates the tradeoff to decrease product costs knowing that the product will not meet a planned functional attribute objective
- More work, more people, more investment: Every time a change is done, it creates rework and requires more people and therefore increased development costs in order to maintain the target project schedule.
- More changes. More investment: The last feedback loop indicates that every time a change is done, it usually increases the investment due to tool modifications or new tools. It is rare to find an opportunity that decreases investment, and they are usually achieved through reusing designs instead of designing new parts.

Figure 5.2 also indicates that the project schedule has a negative relationship with the number of changes, because late changes generate schedule overruns, as also validated by Siddiqi et. al (2011). This correlation is achieved through design freezes as they do not allow the optimization cycles (cost reductions, tough choices or attributes improvement) to continue generating further changes.

Another conclusion about diagram 5.2 first is that product investment is the hardest objective to achieve, since the design optimization actions create incremental changes and these changes usually represent additional investment either for modifying production tools or by requiring additional work to complete. The only real opportunity to improve significantly the investment is by reusing designs to avoid the incremental investment driven by the new parts.

Finally, figure 5.2 indicates that there are more reinforcement loops than balancing loops to attempt to balance the project's iron triangle and the different shortfalls for attributes or cost will only be solved by acknowledging that the project is infeasible with the indicated objectives and they have to be revisited by upper management; nevertheless, this revisiting of objectives often occurs only with a significant delay since the team members will always be tasked to find more efficient solutions and designs and will only be allowed relief if there is enough evidence that all the optimization opportunities have been pursued.

A similar causal loop diagram and feedback loops as in figure 5.2 should be present in project B, however, when common parts are being included, the number of feedback loops doubles because the objectives pressure can come from either of the two projects. The higher number of causes for changes in common parts can be appreciated in the original causal loop diagram in figure 5.1. Therefore, the probability to have a change in a common part is higher than the probability to have a change in a unique part.

To clearly represent the causes and effects of divergence and convergence in the project, the original causal loop diagram in figure 5.1 was simplified to capture the dynamics of divergence and the lifecycle offsets. All the changes driven by all the causes explained above should have an opportunity to be communized (for the unique parts) or a risk of divergence (for the common parts).

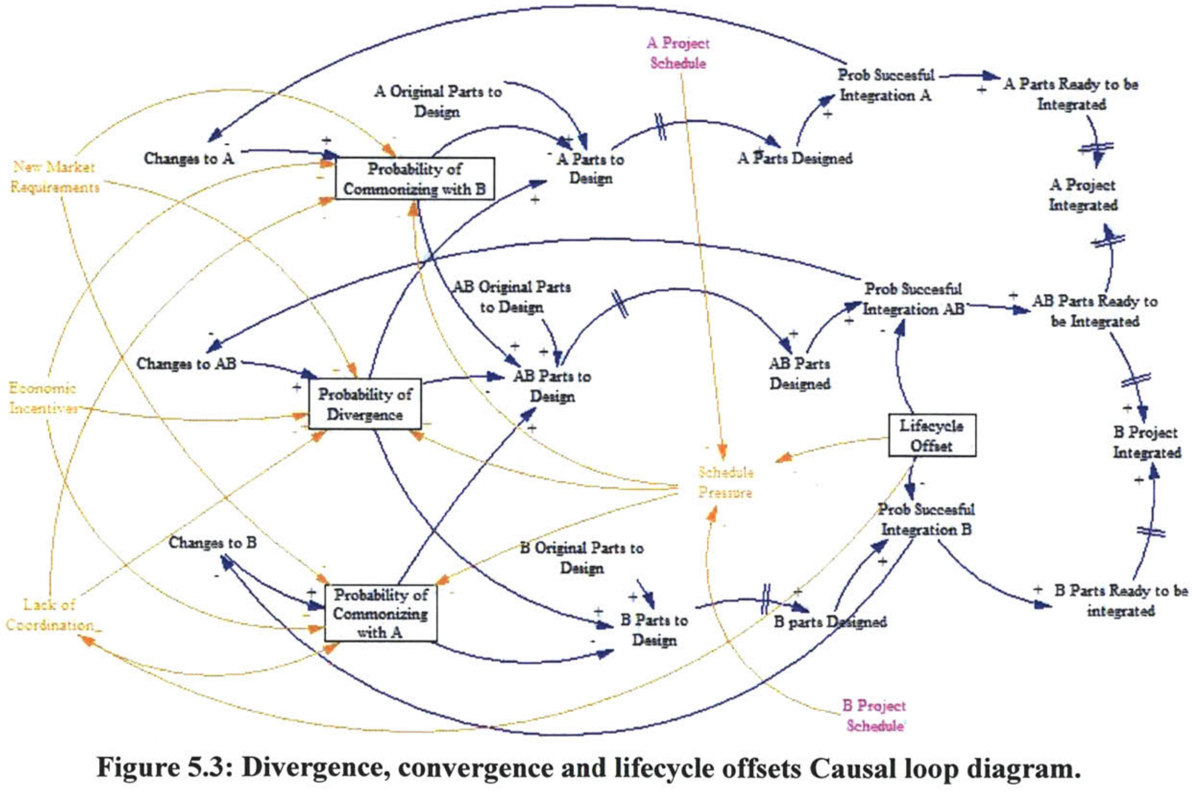


Figure 5.3: Divergence, convergence and lifecycle offsets Causal loop diagram.

As extensively addressed in this thesis, the probability of divergence is always higher than the probability of convergence, noted as the different probabilities of communizing in figures 5.1 and 5.3. This was also found in the case study as exemplified in figure 4.33, where the fraction of changes involving convergence represented barely 10% of the changes, whereas divergence was found in 40% of the changes. These probabilities are affected by apparent external factors, but some of them can be addressed by the development team. All these factors affect inversely commonization and divergence as illustrated in figure 5.3. The following factors were found to affect divergence (and convergence):

- Lack of coordination. With a higher lack of coordination, divergence is more likely and convergence is less likely to occur. Furthermore, the lack of coordination was found to be driven by the structure of the team or the project hand-off.
- Economic incentives. When economic incentives such as shared manufacturing lines are present, there is more willingness to commonize parts and more pressure to avoid divergence. These incentives are usually driven by the overall product and project strategy or by the supplier selection.
- New market requirements. Market requirements can change over the progress of the project, and these changes are more likely to end in divergence than convergence.

If the arrows are followed for the commonization and divergence factors, it can be noticed that the loop is not closed when a part diverges or converges; therefore, these phenomena represent a scope change for the project and not a feedback effect. When designs diverge, the scope is increased and when designs converge the scope decreases; therefore, in order to improve project performance, product commonality has to be actively managed to ensure a timely and issues-free project execution.

The additional factor that has to be addressed in the framework is the lifecycle offsets, which are also represented in figure 5.1. As seen in the figure, these lifecycle offsets represent a further risk to increase the lack of coordination within the team, but on the other hand, it decreases the schedule pressure for the second project as it should have more time to make a common design work, however, the real effect is in the form of a delay; when schedule pressure is present in the second project, and a common part does not function correctly, then it is likely to be modified for a unique part for the second project in order to avoid disruption of the first project that could affect its schedule.

The causal loop diagrams above explain the causes for the various changes to the product through its development; however, these were not included in the simulation model in the prior chapter based on the availability of information and because the model does not attempt to explain the causes for design changes and divergence, but their effect on the product development projects. The models were structured in a way that design changes appear as an external factor, but the actual number of design changes is correlated to the case study, at least the changes that required some tradeoff between functional attributes, cost and schedule that were addressed in the change control meetings with the platform project manager. In addition, the divergence rate was also measured and incorporated into the project, however, the causes of divergence are not included in the model and are therefore treated as an external factor.

Figure 5.4 illustrates the design rework cycle, whose same structure was duplicated for the common parts and the unique parts for the second project. The figure illustrates the most important feedback loops to maintain the project under control and is relevant to illustrate some countermeasures to keep the project schedule. As illustrated in the figures, at the center of the rework cycle, the additional number of changes causes more parts to enter the rework cycle instead of being designed correctly the first time. As said before, the number of changes was correlated with two different methods by estimating the total number of changes of a part based on prior projects experience or by actually estimating the total number of changes based on the change control meeting minutes.

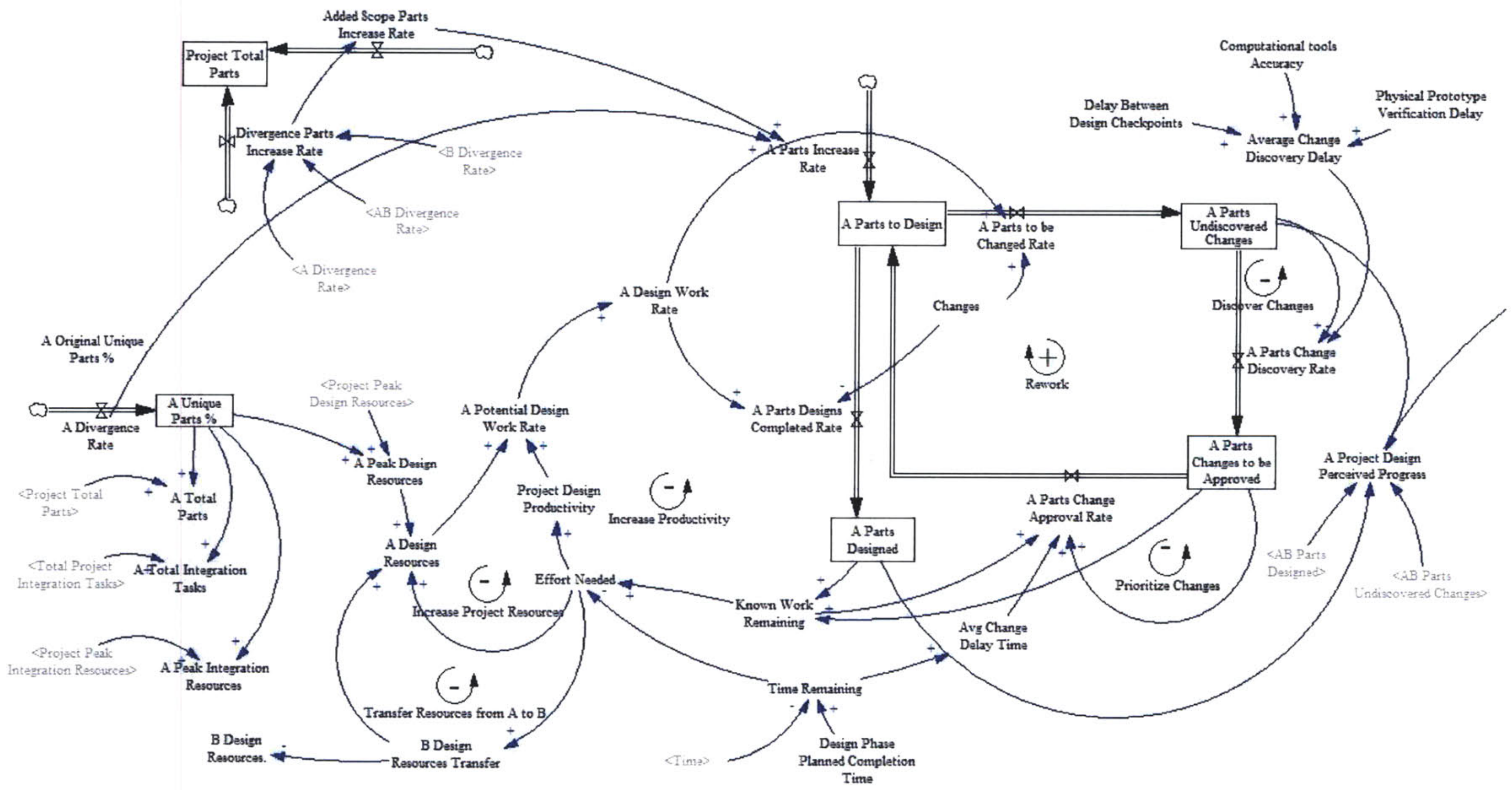


Figure 5.4: Causal loop diagram of the design rework cycle and the countermeasures to keep the project under control.

The figure illustrates four different countermeasures to minimize the impact of changes and to achieve the project schedule:

1. Increasing project resources (for common and unique parts) based on the total number of parts to be designed
2. Transferring budgeted resources from one project to other in order to maintain at least the schedule of the lead project.
3. Increasing the productivity of the design resources, this can be achieved by working overtime or by working faster (cutting corners) but understanding that this may have a knock on effect on the total number of later changes, which is not captured in the model.
4. Discovering changes earlier. This can be achieved by reducing the average design cycle time with more people integrating the product, or by improving the confidence level of the virtual development tools to discover potential design issues earlier. Also, building prototypes earlier will allow earlier feedback to the engineers to discover the design issues earlier.
5. Prioritizing changes. The model structure included a feedback loop based on the behavior of the change control meeting when the design freeze approached, the number of changes increased. By reducing the processing time of the changes, the rework cycle can be accelerated.

The integration rework cycle was modeled to be dependent on the design rework cycle as developed in the framework of the design and integration work dependencies developed in the prior chapter. The integration rework had a similar structure and similar countermeasures are useful. In this case the only difference is that there is no additional delay based on the change control meeting, but only a small delay for some errors that can be found during the project integration.

Some of these five potential countermeasures will be investigated using a hypothetical project to understand the feedback effects and validate these managerial actions in order to keep the project schedule. In addition, these will be compared with the actual countermeasures that were included into the system dynamics model.

5.2 Case Study Iron Triangle Trade-offs: Schedule, Performance and Cost.

The prior two chapters demonstrated that all the different projects experienced divergence throughout their development, regardless the management support for product families and pressures towards design commonization. As explained earlier, product commonality is a way to achieve lower costs and a measure of design efficiency and should not become a blind objective for the project manager. During the development, several design changes were approved and in general, all these changes represented a tradeoff for the project. Using the meeting minutes, the different tradeoffs were analyzed and classified into the following seven categories based on the feedback loops driven by the product optimization cycle and some of the drivers of divergence. The relative relevance for each factor is illustrated in figure 5.5:

1. Functional Attributes: Selecting an attribute over lower costs or commonality.
2. Commonality: Achieving common designs or avoiding divergence assuming that the parts can be overdesigned in terms of functional performance or cost.
3. Cost Reduction: Achieving lower costs through the engineering change process or rejecting expensive changes that do not improve significantly the overall product design or performance.
4. Feasibility: Required changes to achieve the minimum expected quality without any possibility of trade off. In most of the situations these changes are required to sell the product.
5. Market: Changes required to achieve marketing wants, usually traded off for higher product costs
6. Timing: Achieving the project timing and targeted completion schedule
7. Tough Choice: Giving up functional attributes or market changes to achieve lower costs

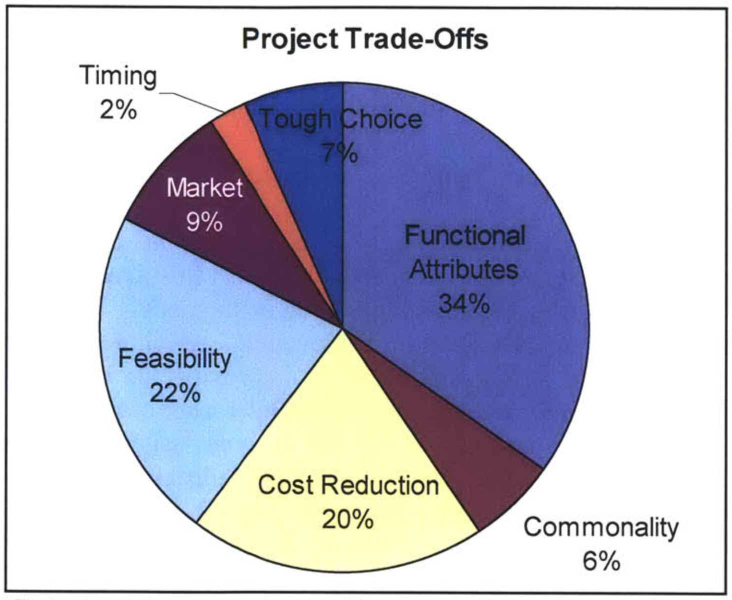


Figure 5.5: Relative relevance of the different optimization actions and changes

The project was closely monitored by upper management and the project managers had a considerable pressure to achieve overall product costs in terms of production, development and investment costs. This pressure should have suggested that the project manager should have selected usually the lower cost alternative in most of the product changes, however, as seen above lower costs (through cost reductions or tough choices) represented only 27% of the choices, compared to the 43% frequency where achieving the project scope (functional attributes or market wants) was preferred over cost reductions.

While the data above appears to demonstrate that the priority for the project manager was to achieve the expected product performance and quality, there are other factors that may have driven the choices above, even with the high management pressure to achieving lower costs. The trend to accept the tradeoff of higher costs for better performance is based on the nature of each of the alternatives. In general, there are more ways to achieve lower overall product costs than to achieve the expected product performance. Consider for example achieving the range requirement of the vehicle; two main options appear: either increasing the tank volume or improving fuel economy; on the other hand, achieving lower costs can be materialized with low cost country sourcing, using less expensive materials, further concessions in commercial negotiations, etc. In general, the cost of the product can be addressed by any of the components of the product; however, the engineering issues usually require a specific set of components that can modify the product behavior.

A hidden effect in the data above is the iron triangle tradeoff for achieving the project schedule. Because the data above represents the initial phases of the product development process and not the complete project, the effect of the schedule pressure has vanished. Yet, during project development, some examples of the schedule for cost or performance tradeoffs were presented, and whenever there was a development risk for not achieving the project schedule, these clear opportunities were not pursued. Two examples were significant that represented this tradeoff in the case study, and these were the development of a new power management strategy and the commonization of a crossmember. Both were rejected because the projects could not accommodate all the design changes required, even if there was still plenty of time to incorporate them.

A final decision mode or reflection of priorities that were included in the framework above were the decisions and changes made in order to maintain or improve project commonality. The case study explicitly demonstrated the importance of product commonality, and it was backed up with a product plan and senior management that also supported the benefits of product commonality and product platforms. These decisions and changes were made in order to keep the designs common representeded 6% of all the changes. While this represents a small fraction of all the changes, these represent the convergence opportunities that were pursued and introduced into the project plan and driven by the project manager and the project structure that emphasized product commonality; however, these were significantly dominated by other project priorities like product costs and product functionality.

In conclusion, the iron triangle priorities of the platform project manager in the case study followed the following hierarchical order:

1. Project Schedule: This was the highest priority and achieving the project schedule would became an even higher priority in more advanced phases of development.
2. Product performance: When a basic quality issue was present, there were few opportunities for a tradeoff; however, when a performance quality issue was present, the nature of the improvement was traded off with the costs and either pursued or rejected.
3. Product/project costs: This represented the lowest priority based on the data above; however, all the changes usually represented higher costs, without adjusting the cost objectives accordingly.

5.3 Complete Project and Partial Project Performance – Baseline Behavior

In chapter 4, a system dynamics model was developed and calibrated to accurately represent the actual project behavior using measured variables like divergence rate, the total number of parts and original product commonality. In the present section, the same system dynamics model will be used with a hypothetical case study to simplify the project behavior and provide a tool to quantify the relative impact of all the relevant variables in the model.

The hypothetical model was selected to have equal scope for both products (same number of parts) and the divergence rate for the common parts was balanced with two equal convergence rates for the unique parts for project A and project B. Divergence still continued to be modeled as a linear reduction in the percentage of common parts, and a linear increase in the number of unique parts. These hypothetical projects are summarized in table 5.1.

Design Stage	Expected Completion time (weeks)	% of tasks to be accomplished (weeks)	Integration Original Work To Do [Tasks]		Total parts to design (Parts)			Peak Resources [People]	
			Design	Integration	Unique Parts A	Common Parts	Unique Parts B	Design	Integration
Concept Development	24	8.48%	1320.7	931.8	1000	1000	1000	155.3	109.6
Detailed Design	64	31.41%	4891.8	3451.4					
Testing and Refinement	100	59.41%	9252.5	6528.1					
Complete Project	172	100%	15573.9	10988.2					

Table 5.1. Hypothetical Project input variables.

In addition other parameters were also unchanged from the baseline case study. Furthermore, the productivity of the integration resources also followed the behavior explained in figure 4.26 where productivity increases as the project phase deadline approaches, and the behavior of prioritizing the pending engineering changes explained in figure 4.23 remained unchanged. Also all the different delays were kept unchanged to allow a reasonable comparison with this hypothetical project. The hypothetical project will be used to cancel the effect of the scope increase. To estimate the parts increase related to product divergence, different parts increase rates were derived for different divergence rates and these are detailed in table 5.2 which was also used as an input to the simulations. Other relevant input variables to the model are detailed in table 5.3.

Divergence Rate (% Common Parts/Week)	Parts Increase Equivalence per 1000 total project parts (Parts / Week)
0.30%	3.49
0.28%	3.23
0.26%	2.96
0.24%	2.71
0.22%	2.46
0.20%	2.21
0.18%	1.97
0.16%	1.73
0.14%	1.50
0.12%	1.27
0.10%	1.05
0.08%	0.83
0.06%	0.62
0.04%	0.41
0.02%	0.20
0.00%	0.00

Table 5.2: Divergence rate to parts increase rate equivalence.

Design Stage	Expected Completion time (weeks)	% of tasks to be accomplished (weeks)	Design Productivity (Parts/week)	Design Churn (Changes per part)	Probability of a flawless design	Integration Quality
Concept Development	24	8.48%	3.635	0.698	0.589	0.8
Detailed Design	64	31.41%	2.617	1.860	0.350	
Testing and Refinement	100	59.41%	2.162	2.907	0.256	
Complete Project	172	100%	2.209	5.000	0.167	

Table 5.3: Additional simulation parameters for the hypothetical project.

A minor update was done to the model to allow a variable discovery delay instead of averages used in the prior model; this will allow the model to easily simulate the complete project. The update did not change significantly the integration completion time, but had some impact on the design completion time. Nevertheless, the project completion time is being measured when the integration activities complete the project.

Table 5.4 details the simulation results for the baseline case study. As seen in the table, most of the phases are expected to have a delay on their completion, including the project completion time; however, it also shows that the design completion times improve significantly after the prototype is built as it allows having shorter times to discover rework. Figures 5.6 to 5.10 provide more details on the behavior of the project.

Design Stage	Expected Completion time (weeks)	% of tasks to be accomplished (weeks)	Simulation Results				
			A Design Time w Common Parts [Week]	B Design Time w Common Parts [Week]	AB Design Time [Week]	A Integration Time [Week]	B Integration Time [Week]
Concept Development	24	8.48%	23.18	23.1875	23.1875	30.56	30.56
Detailed Design	64	31.41%	81.25	81.25	81.25	68.93	68.93
Testing and Refinement	100	59.41%	115.68	115.68	115.68	104.06	104.06
Complete Project	172	100%	172.56	172.56	172.56	179.31	179.31

Table 5.4: Hypothetical project baseline behavior for the four design phases.

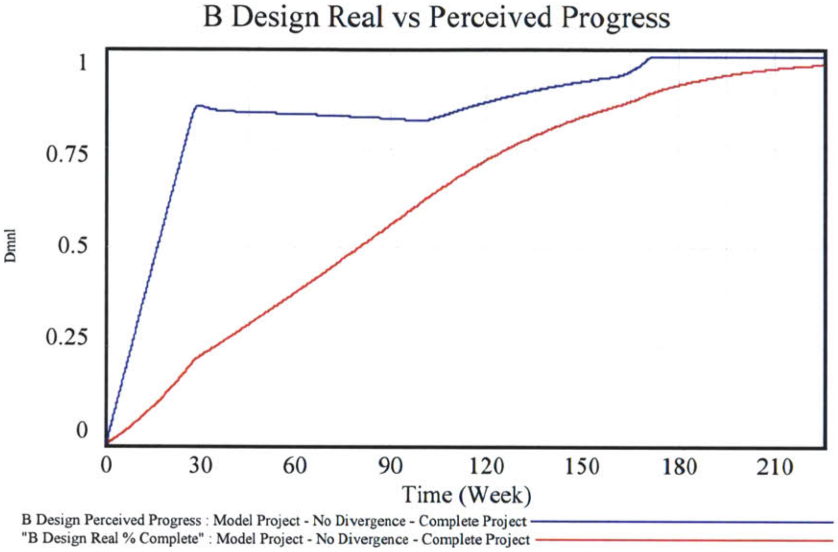


Figure 5.6: Real vs perceived progress.

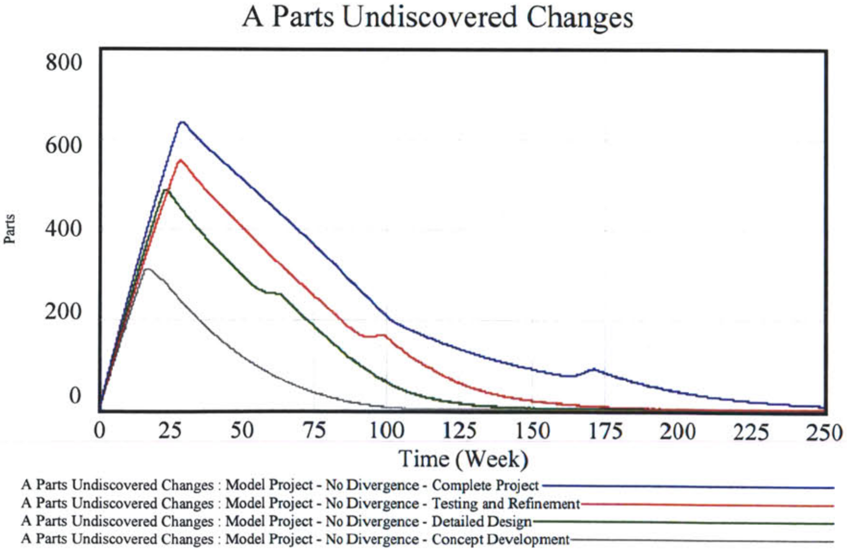


Figure 5.7: Undiscovered changes behavior.

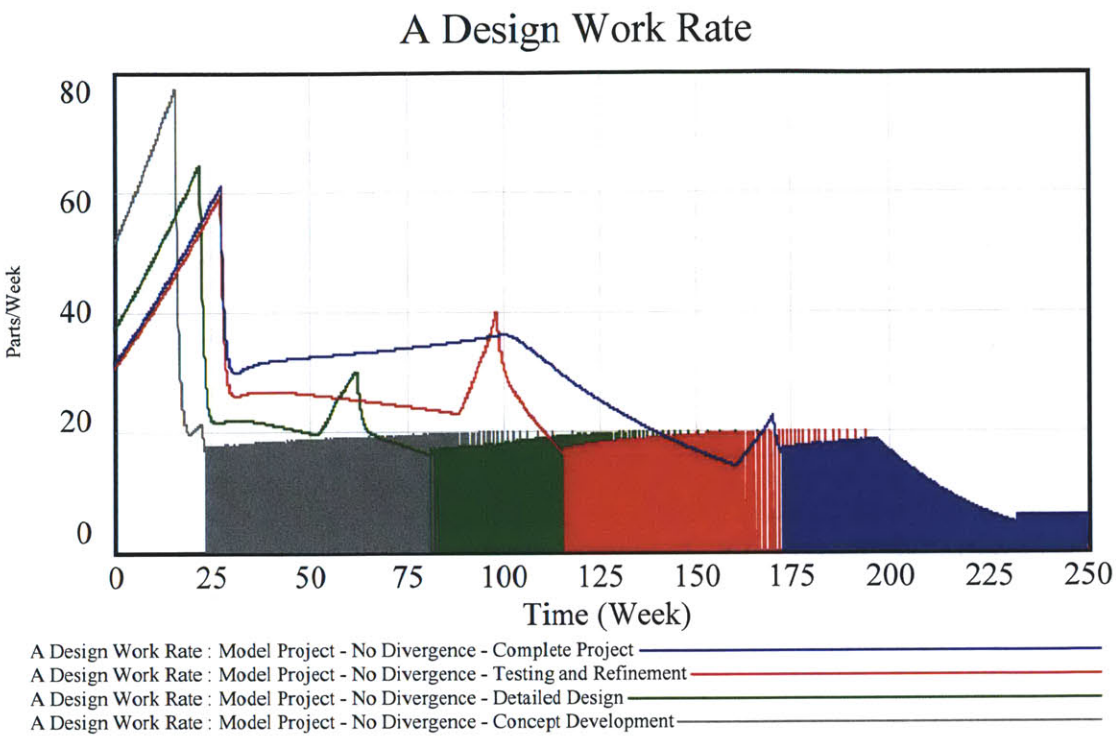


Figure 5.8: Design work rate behavior

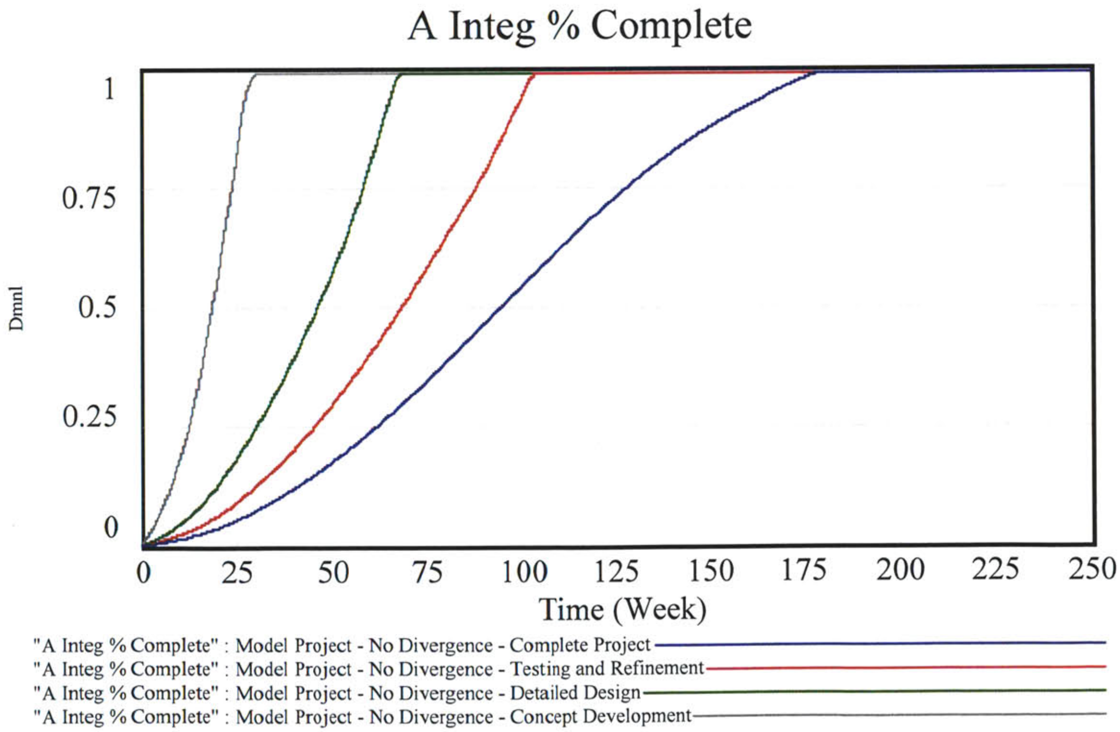


Figure 5.9: Integration Completion behavior

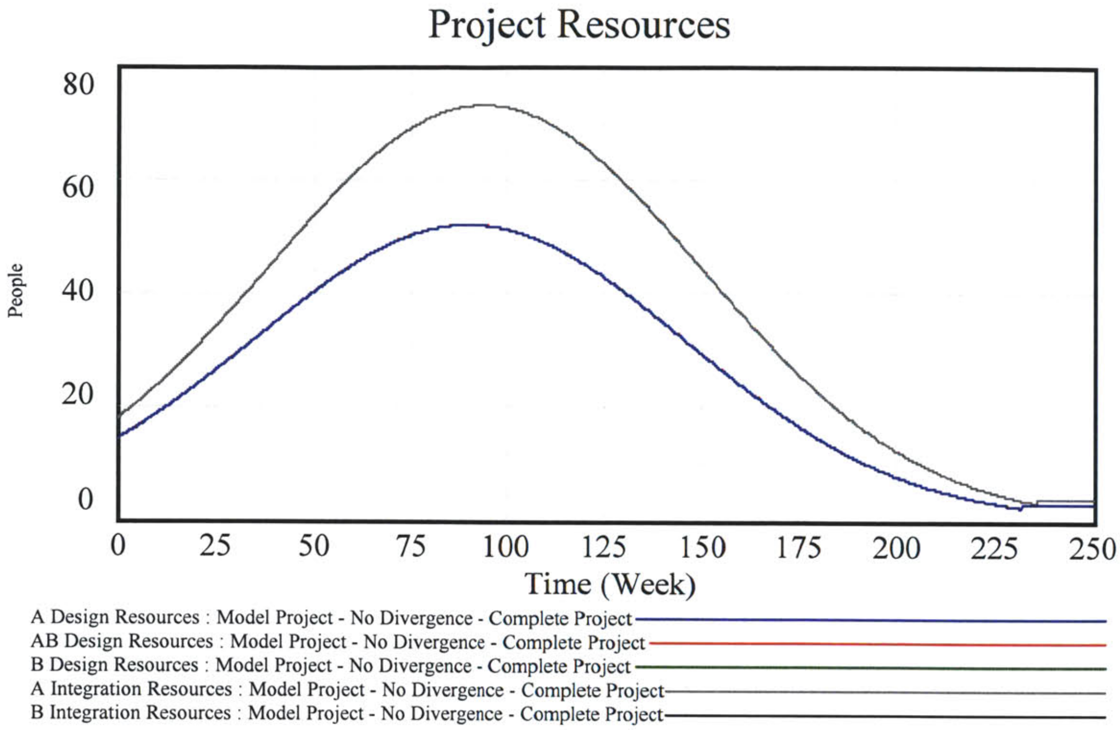


Figure 5.10: Design and integration resources

The figures above (5.6 to 5.10) represent some of the Vensim ® software output curves. Figure 5.6 show the perception gap between the actual progress and the perceived progress. The project overestimates the progress at the beginning of the project and then starts a phase of no apparent progress. The reason for this is that the number of undiscovered changes has increased significantly as shown in figure 5.7. As the project progresses, the team will find these changes to be done in order to achieve a feasible and compatible product.

Figure 5.8 represents the behavior of the design work rate. Similar to the perceived progress, during the beginning of the project the rate increases proportional to the headcount curve, however, then the work rate is reduced significantly because apparently there is no more work to be done, although, the work to be done will be dependent on the discovering the required changes. The work rate also experiences a second peak, based on the feedback loop of prioritizing the changes.

Figure 5.9 shows the progress for the integration resources, which is modeled to be dependent on the progress of the design work and proportional to the number of human resources. The progress has a steady increase throughout all the phases. In addition, figure 5.10 details the headcount curve for the design and integration resources. Because the project has the same amount of parts for both products, the curve is the same for all.

5.4 *Effects of Divergence and Lifecycle Offsets on Project Schedule and Overall Workforce*

Once the project baseline behavior was understood, the case study and the literature research concluded that product commonality decreases over time and that this is a characteristic of the development of complex products. The prior baseline scenario will be simulated including the effect of various magnitudes of divergence rates, which will also be further investigated including different lifecycle offsets scenarios.

Divergence was modeled as a linear decrease of the percentage of common parts, and therefore a linear increase of the percentage of unique parts for the other two projects. Since the divergence data was investigated during the design period of the project, the model will assume that divergence ends after the design phase. This assumption may not be adequate since a significant portion of changes are still required to the product, and an asymptotic behavior could be more appropriate in this case. This is a model limitation based on the information available, however, the model will be incorporating the effect of the loss of commonality and this will further be investigated with potential managerial actions to overcome it.

Lifecycle offsets were modeled as a delay in the start (and finish) time of the project, proportional to the offset. The project is expected to start and finish later, and its resources will be deployed with a similar headcount curve. The model will not explicitly include the effect of development time improvement for the second derivative; however, it is a common business practice and expectation in product family development. In addition, the lifecycle offsets effect will be present also in common parts development, which will be modeled as a further delay of the time to discover rework, as the common parts will be entirely validated until the second derivative completes its development.

The system dynamics model will now be used to simulate complete projects instead of the various design phases used to calibrate the model as was included in the prior section. The intent will be to forecast the end result of divergence on the development project and its effect with different lifecycle offsets. Figure 5.11 illustrates the effect of divergence on project completeness for various divergence rates ranging from no divergence to 0.3% loss of common parts per week.

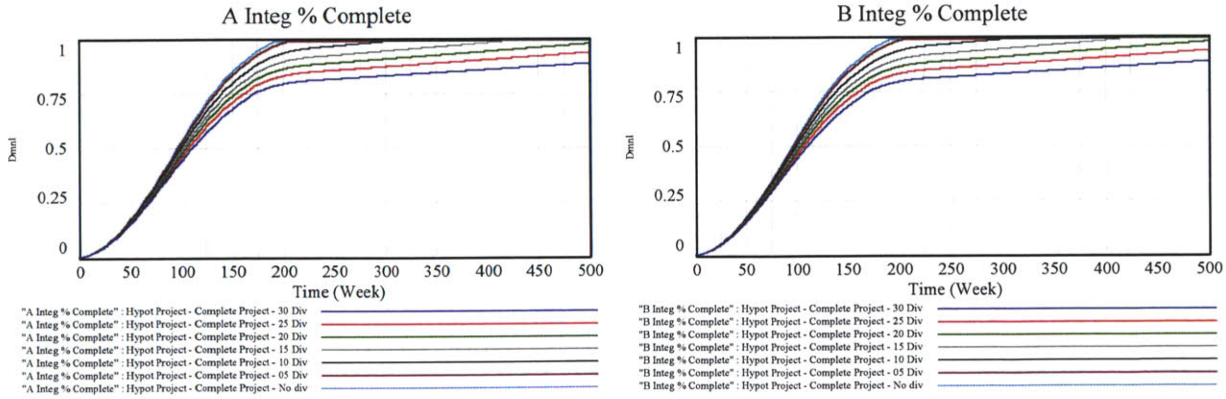


Figure 5.11: Effect of divergence on project completion for project A and project B.

As seen in the figure below, divergence creates a non-linear effect on project completion, and most of the non linearity is based on the effect of the long tail of the headcount curve. As the project approaches its conclusion, the number of resources assigned is also reduced and therefore the progress is marginal based on the remaining available people. In order to eliminate the effect of the long tail, a new set of simulations were performed using a constant headcount based on the same engineering hours of the headcount curve. Figure 5.12 illustrates the clear effect of various divergence rates to the project completion time when a constant headcount is assumed, these are summarized in table 5.5 below.

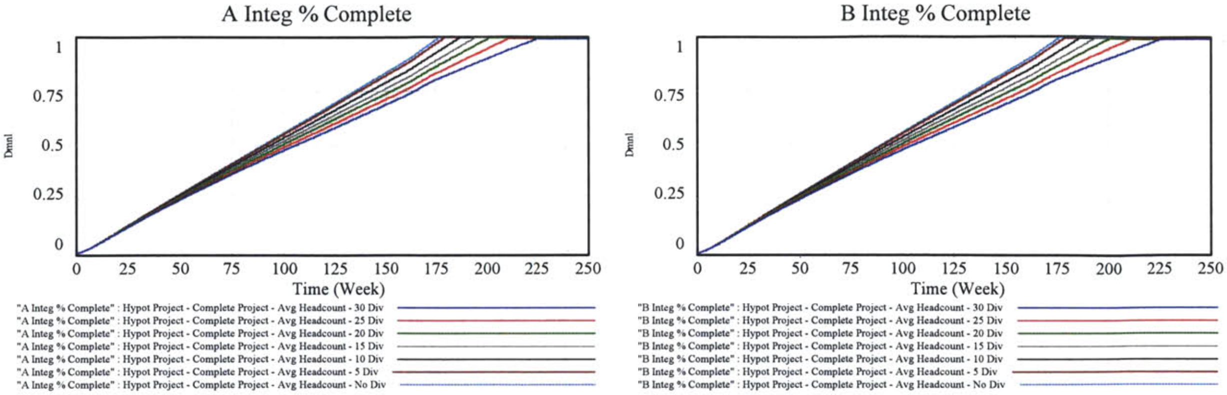


Figure 5.12: Effect of divergence on project schedule with an average headcount.

Divergence rate (% common parts / week)	Expected Completion Time (Weeks)		Simulated Completion Time (Weeks)		% of Schedule Overrun compared to baseline without divergence	
	Project A	Project B	Project A	Project B	Project A	Project B
No Divergence	172	172	171.188	171.25	Baseline	Baseline
0.05%			176.93	176.93	3.35%	3.32%
0.10%			182.87	182.94	6.82%	6.83%
0.15%			189.12	189.18	10.48%	10.47%
0.20%			195.56	195.62	14.24%	14.23%
0.25%			202.313	202.37	18.18%	18.17%
0.30%			209.25	209.31	22.23%	22.22%

Table 5.5: Expected schedule overrun driven by divergence.

In addition, the lifecycle offsets were included into these simulation to estimate the project completion time in two scenarios for a 12 weeks and a 24 weeks offset. The results of these simulations are shown in table 5.6 which illustrates that project completion time is not dependant on the lifecycle offsets, and for the greater the divergence rates, the greater the schedule overrun. In fact, the follow on projects experienced better than expected completion times, reflecting the advantage of designing the common parts earlier in the follow-on derivative project.

Divergence rate (% common parts / week)	Expected Completion Time (Weeks)		Simulated Completion Time (Weeks)		% of Schedule Overrun compared to baseline without divergence	
	Project A	Project B	Project A	Project B	Project A	Project B
No Divergence	172	184	171.12	178.56	Baseline	Baseline
0.05%			176.87	184.56	3.36%	3.36%
0.10%			182.81	190.75	6.83%	6.83%
0.15%			189.06	197.31	10.48%	10.50%
0.20%			195.5	204.06	14.25%	14.28%
0.25%			202.25	211.18	18.19%	18.27%
0.30%			209.25	218.56	22.28%	22.40%
No Divergence	172	196	171.12	186.5	Baseline	Baseline
0.05%			176.87	192.87	3.36%	3.42%
0.10%			182.81	199.43	6.83%	6.93%
0.15%			189.06	206.37	10.48%	10.65%
0.20%			195.5	213.5	14.25%	14.48%
0.25%			202.25	221.06	18.19%	18.53%
0.30%			209.18	228.93	22.24%	22.75%

Table 5.6: Expected schedule overrun driven by divergence with lifecycle offsets.

To complement the iron triangle framework, if a projected schedule overrun is not acceptable, another alternative is to increase the project resources to increase the work rate and achieve the project schedule. Using the system dynamics model with the project's headcount curve, a new set of simulation runs were performed in order to investigate the equivalent headcount to achieve the target project schedule. These results are summarized in table 5.7:

Divergence rate (% common parts / week)	Expected Completion Time (Weeks)		Simulated Completion Time		Baseline Peak Resources (People)		Required Peak Resources to meet schedule (Weeks)		Additional Resources % to meet scheule	
	Project A	Project B	Project A	Project B	Design	Integration	Design	Integration	Project A	Project B
No Divergence	172	172	171.875	171.875	155.3	109.6	159.18	112.34	2.50%	2.50%
0.05%			172	172			165.75	116.97	6.73%	6.72%
0.10%			171.938	171.938			172.53	121.76	11.09%	11.09%
0.15%			171.938	171.938			179.53	126.69	15.60%	15.59%
0.20%			172.31	172.31			186.36	131.52	20.00%	20.00%
0.25%			171.56	171.56			194.125	137	25.00%	25.00%
0.30%			172	172			201.11	141.93	29.50%	29.50%
No Divergence	172	184	171.938	176.68	155.3	109.6	158.87	112.12	2.30%	2.30%
0.05%			172.125	176.68			165.55	116.83	6.60%	6.60%
0.10%			172	176.43			172.38	121.65	11.00%	10.99%
0.15%			172	176.31			179.37	126.59	15.50%	15.50%
0.20%			172.125	176.188			186.36	131.52	20.00%	20.00%
0.25%			172.06	176			193.65	136.67	24.69%	24.70%
0.30%			171.875	175.93			201.11	141.93	29.50%	29.50%
No Divergence	172	196	171.875	182.44	155.3	109.6	159.03	112.23	2.40%	2.40%
0.05%			172	182.5			165.55	116.83	6.60%	6.60%
0.10%			171.93	182.25			172.383	121.65	11.00%	10.99%
0.15%			171.93	182.06			179.37	126.59	15.50%	15.50%
0.20%			172.06	182			186.36	131.52	20.00%	20.00%
0.25%			172	181.81			193.65	136.67	24.69%	24.70%
0.30%			172	181.62			200.95	141.82	29.39%	29.40%

Table 5.7: Equivalent headcount increase to achieve project schedule.

Tables 5.6 and 5.7 summarize in two different measures the end effects of divergence and lifecycle offsets to product family development projects. As it can be appreciated, these effects are significant and are proportional to the relative divergence rate, measured as a percentage of percent commonality loss per period of time. The case study experienced divergence rates as high as 0.3% commonality loss per week, indicating that the rates above can be found in industry. On the other hand, these results also indicate that lifecycle offsets do not affect negatively the performance of the project, furthermore, lifecycle offsets improved the expected development time. The causes of this improvement are the common parts that are effectively designed that will be reused without further changes. However, excessive lifecycle offsets can cause larger divergence rates caused by lack of coordination between the projects.

In addition, when long lifecycle offsets are found, it can also create a loss of experience and knowledge in the workforce. This knowledge dilution can be caused by team members moving from one project to other, the natural hire and fire rate in the company or by the lack of documentation of the development process.

5.5 Managerial Recommendations for Improved Performance of Product Families Projects Schedule

The prior section used a hypothetical project to quantify the effects of divergence. The present section will use the same model to illustrate the effect of proposed managerial actions to keep the project within its objectives for cost, scope and schedule. The literature review in product platforms has extensively addressed the financial and product performance topics on product platforms, however, little research has been done on the effect of development time for both the lead and the subsequent derivative projects. As found in the literature review, project schedule was the only characteristic that was unlikely to be traded off, therefore all the countermeasures to be discussed will be recommended for keeping the project schedule and all the effects will be studied based on the effect on project schedule. The managerial actions can be classified in two major categories, derived from the rework causal loop diagram in figure 5.4:

1. Countermeasures to manage the project overall without addressing the rework cycle
2. Countermeasures to manage and minimize the rework cycle.

The first set of countermeasures includes the traditional managerial actions that improve the work rate based on project resources. Two alternatives paths are usually followed: increasing work rate by adding more resources or increasing the work rate by improving the relative productivity either by working extra time (i.e. weekends or extra hours) or by working faster (and therefore increasing the probability of not performing the work correctly). All the countermeasures will be quantified by measuring the relative improvement on project schedule, considering an average number of resources instead of the headcount curve.

Table 5.8 summarizes the effect of increasing the productivity of the people working in the project by 5% or 10%. As seen in the table below, increasing productivity positively impacts project schedule which is improved around 4% per each 5% improvement in productivity. The table also shows that an increase in 10% productivity does not offset a 0.15% divergence rate. In addition, the model does not capture knock on effects of increasing productivity that usually decrease the overall quality of the work and increase the probability of errors in the integration phase and generate more changes in the product.

Relative Improvement	New parameter inputs	Divergence Rate (% comm loss / week)	Lifecycle Offset (Week)	Estimated Completion Time (Week)		Relative % Improvement Compared to Baseline		Average Improvement
				Project A	Project B	Project A	Project B	
Baseline	Design productivity: 2.21 part/wk Integration productivity: 1 task/wk	No Divergence	No Offset	171.188	171.25	Baseline		Baseline
		No Divergence	24 Weeks	171.12	186.5			
		0.15%	No Offset	189.12	189.18			
		0.15%	24 Weeks	189.06	206.37			
		0.30%	No Offset	209.25	209.31			
		0.30%	24 Weeks	209.18	228.93			
5% Improvement	DP: 2.32 part/week IP: 1.05 task/wk	No Divergence	No Offset	164.5	164.5	3.91%	3.94%	4.10%
		No Divergence	24 Weeks	164.43	178.43	3.91%	4.33%	
		0.15%	No Offset	181.31	181.37	4.13%	4.13%	
		0.15%	24 Weeks	181.25	198.56	4.13%	3.78%	
		0.30%	No Offset	200.18	200.188	4.33%	4.36%	
		0.30%	24 Weeks	200.12	219.875	4.33%	3.96%	
10% Improvement	DP: 2.43 part/week IP: 1.1 task/wk	No Divergence	No Offset	157.37	157.43	8.07%	8.07%	7.94%
		No Divergence	24 Weeks	157.31	171.06	8.07%	8.28%	
		0.15%	No Offset	174.37	174.37	7.80%	7.83%	
		0.15%	24 Weeks	174.25	191.625	7.83%	7.14%	
		0.30%	No Offset	192.06	192.06	8.22%	8.24%	
		0.30%	24 Weeks	192	211.75	8.21%	7.50%	

Table 5.8: Effect of increased productivity on the project schedule.

An alternative way to increase the work rate without affecting the quality of project execution is by adding additional resources to the product to achieve the project schedule. The effect of adding these incremental resources has been already addressed in the prior section by estimating the number of resources required to achieve project schedule. If resources are increased 5% or 10% as in the prior assessment, the effect on project schedule would be the same and is therefore omitted from this study. The difference between adding resources and increasing productivity is that adding resources will inevitably increase project's costs while increasing productivity may incur lower costs but can affect the quality of the work being done.

The third alternative affecting the work rate is to counterbalance the effects of divergence by shifting resources from one project to the other to prioritize the schedule of the first project. Said more clearly, staff is transferred temporarily from project B to project A so that Project A can meet its project completion target. This is done at the expense of the schedule of Project B. In the case study, this countermeasure was common as the project manager prioritized the work to be done based on deadlines, and usually the lead project was prioritized over the follow-on projects. The project manager emphasized the relevance of the first project as several parts were planned to be reused and the progress done on common parts impacted both projects equally. Table 5.9 details the effect on the schedule of both projects when resources were shifted from one project to other, and figure 5.13 further illustrates the behavior of the derivative project and its resources.

Relative Improvement	Divergence Rate (% comm loss / week)	Lifecycle Offset (Week)	Estimated Completion Time		Relative % Improvement	
			Project A	Project B	Project A	Project B
Baseline	No Divergence	No Offset	171.188	171.25	Baseline	
	No Divergence	24 Weeks	171.12	186.5		
	0.15%	No Offset	189.12	189.18		
	0.15%	24 Weeks	189.06	206.37		
	0.30%	No Offset	209.25	209.31		
	0.30%	24 Weeks	209.18	228.93		
With Shared Resources	No Divergence	No Offset	171.375	171.43	-0.11%	-0.11%
	No Divergence	24 Weeks	171.25	181.62	-0.08%	2.62%
	0.15%	No Offset	175.75	207.93	7.07%	-9.91%
	0.15%	24 Weeks	175.62	217.62	7.11%	-5.45%
	0.30%	No Offset	177.81	243.688	15.03%	-16.42%
	0.30%	24 Weeks	177.68	252.188	15.06%	-10.16%

Table 5.9: Effect of project prioritization and shifting resources from one project to other.

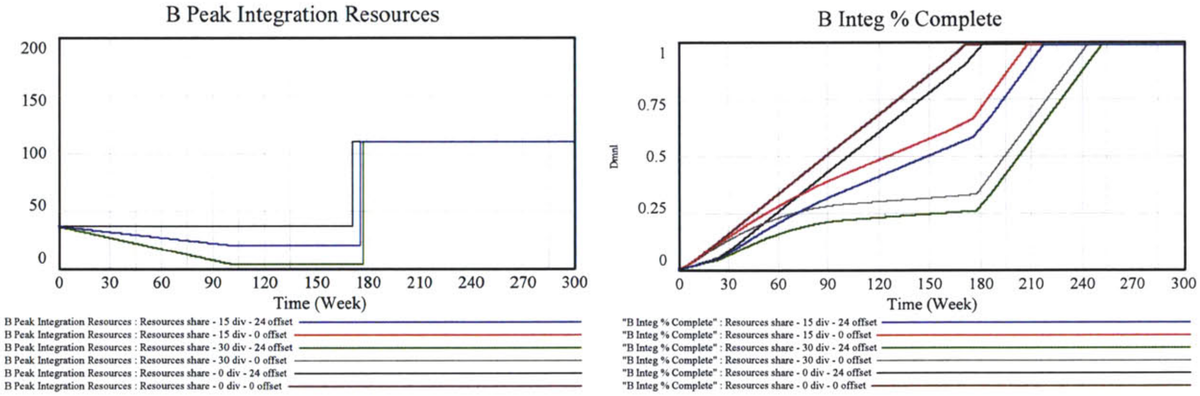


Figure 5.13 Effect of shifting resources from one project to other. The first figure shows the allotted resources to the second project and the second shows its effect on overall progress.

The results above illustrate that even if the people are shifted from one project to the other in order to overcome the loss of commonality, the end result can be a lead project finished on time, however, the lack of attention to the second project will end in a significant schedule overrun as seen in the figures above as well as measured in table 5.9. Shifting resources can be a beneficial tool for the project manager if the attention can be shifted temporarily, without disrupting the progress of the second project too much.

System dynamics models emphasize the effect of rework in product development projects. The three recommendations above can improve the project schedule; however, these actions do not address the causes of the rework cycle, which is the main driver for schedule and costs overruns in a wide range of projects (Lyneis & Ford, 2007). Instead of focusing on working harder and faster, additional recommendations are investigated. These recommendations are aimed to address the variables that drive the behavior of the rework cycle, and these can be visualized in figure 5.4 which details the rework cycle causal loop diagram: rework (reinforcement loop), discovery of changes (balancing loop) and prioritization of changes (balancing loop).

The first recommendation based on the causal loop diagram is reducing the overall number of changes. The case study suggested that the average number of changes per part was around 5 changes per each part, meaning that the vehicle will have to be designed once, and then redesigned 5 times. While these changes may be driven by initially infeasible designs that have to be corrected, or by design errors found during the design process; many of these changes may have been caused by attempts to optimize the product. In addition, further changes may have been driven by the integral architecture of the product and the change propagation framework (Eckert et. al. 2004, Giffin et al. 2009).

Table 5.10 shows the relative effect of the total number of changes per part. As seen in table below, reducing the number of changes 5% or 10% does not change the project schedule which is driven by the perception that there are no further changes, but these may occur even after the product starts its production. This can be appreciated in figure 5.14 which details the real design progress for each of the three different rates of change. The different number of changes per part marginally improved the real design progress; therefore, focusing on avoiding changes will not impact the project performance.

Relative Improvement	New parameter	Divergence Rate (% comm loss / week)	Lifecycle Offset (Week)	Estimated Completion Time		Relative % Improvement		Average Improvement
				Project A	Project B	Project A	Project B	
Baseline	Changes per part: 5	No Divergence	No Offset	171.188	171.25	Baseline		Baseline
		No Divergence	24 Weeks	171.12	186.5			
		0.15%	No Offset	189.12	189.18			
		0.15%	24 Weeks	189.06	206.37			
		0.30%	No Offset	209.25	209.31			
		0.30%	24 Weeks	209.18	228.93			
5% Improvement	Changes per part: 4.75	No Divergence	No Offset	171.188	171.25	0.00%	0.00%	0.00%
		No Divergence	24 Weeks	171.125	186.5	0.00%	0.00%	
		0.15%	No Offset	189.125	189.188	0.00%	0.00%	
		0.15%	24 Weeks	189	206.313	0.03%	0.03%	
		0.30%	No Offset	209.25	209.313	0.00%	0.00%	
		0.30%	24 Weeks	209.188	228.93	0.00%	0.00%	
10% Improvement	Changes per part: 4.5	No Divergence	No Offset	171.188	171.25	0.00%	0.00%	0.00%
		No Divergence	24 Weeks	171.125	186.5	0.00%	0.00%	
		0.15%	No Offset	189.125	189.188	0.00%	0.00%	
		0.15%	24 Weeks	189.06	206.313	0.00%	0.03%	
		0.30%	No Offset	209.25	209.31	0.00%	0.00%	
		0.30%	24 Weeks	209.18	228.93	0.00%	0.00%	

Table 5.10: Effect for reduced number of changes.

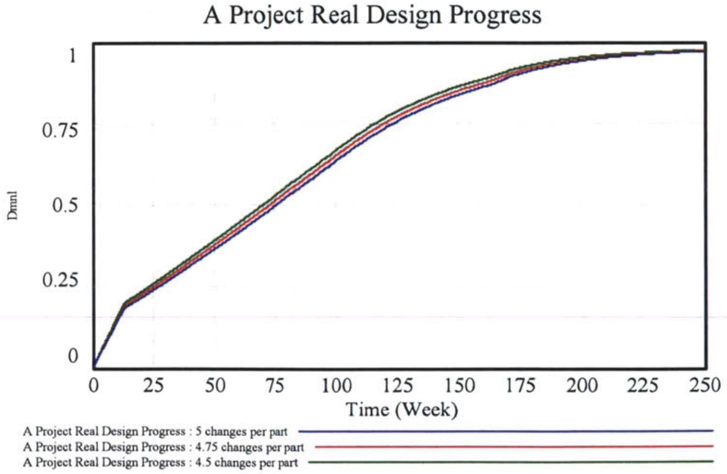


Figure 5.14: Real design progress. Even if the project appears to be completed, the real progress suggests that there are further changes that can happen to the product after production starts.

Similar investigations were done to estimate the improvement in project schedule driven by the rest of the proposed improvements in the rework cycle:

- Scheduling the engineering prototype 5 weeks and 10 weeks earlier (figure 5.15)
- Improving the assessment confidence of virtual and engineering models to avoid future design issues (figure 5.16)
- Reducing the average time between the engineering checkpoints (figure 5.17)

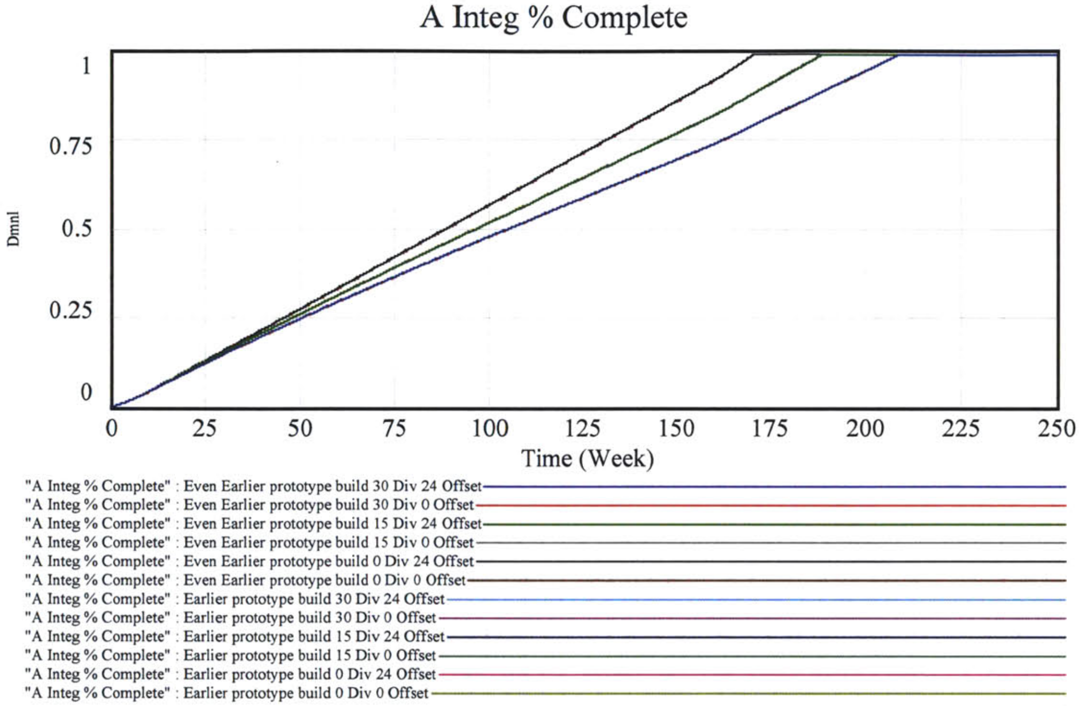


Figure 5.15: Effect of scheduling earlier the engineering prototype.

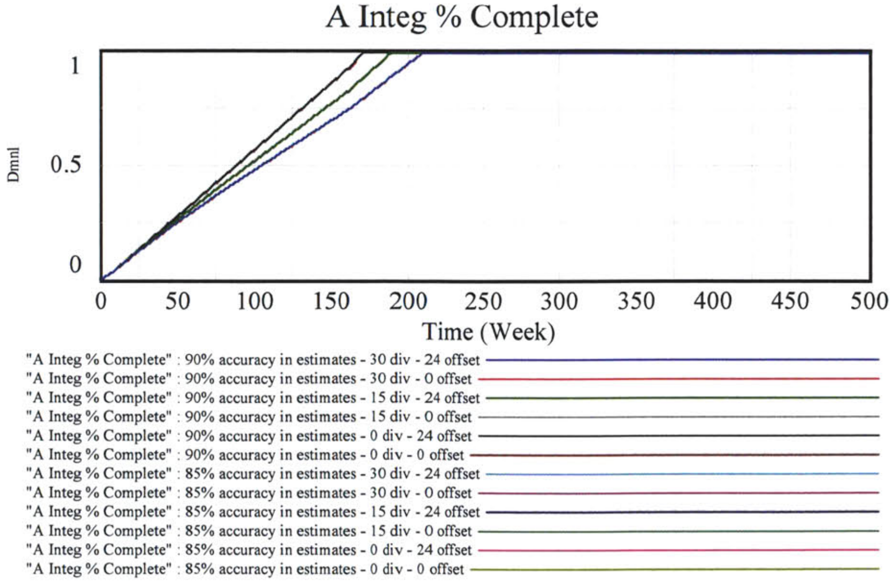


Figure 5.16: Effect of improving the virtual and analytic assessment tools confidence.

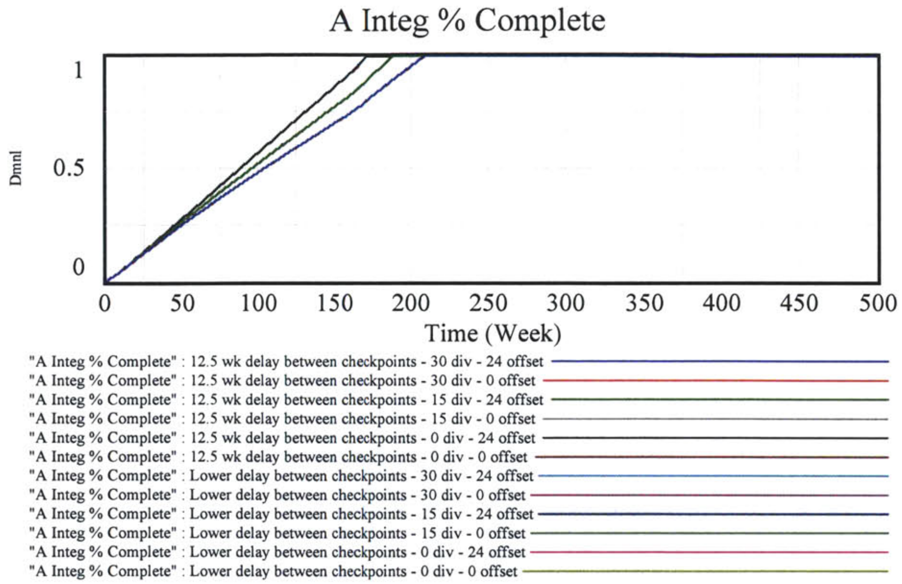


Figure 5.17: Effect of reducing the average time between engineering checkpoints.

The results of these improvements, similar to reducing the total number of changes, did not yield a very signficant improvement in project time based on the system dynamics model assessment. In general, all these actions should contribute an incremental improvement to achieve project's schedule. Unfortunately, the large amount of engineering changes overshadowed the effect of these improvements; however, they should be also considered as part of the set of improvements.

In summary, the set of managerial recommendations above further emphasize the reality of the project iron triangle framework. The recommendations based on typical managerial actions and not based on knowledge of the rework cycle always resulted in a tradeoff between the quality of the work performed and the cost of the project. On the other hand, actions that cancel or significantly mitigate the effects derived from the rework cycle and its delays did not result in a project schedule improvement, driven by the extremely high churn of the design. The effect of high divergence rates, which were measured and found in specific products of the product family, can result in either a 22% schedule overrun or a 29% headcount increase if the schedule is required to be maintained, concluding that focusing on avoiding divergence can be one of the best countermeasures to keep the project under control. If however divergence is found to be necessary to maintain competitiveness of the product in the marketplace then some amount of divergence should be allowed.

5.6 Additional Case Study and Model Findings and Recommendations for Future Projects

System dynamics models proved to be a valuable tool to validate the framework of divergence and understand the potential impacts of divergence on a product development project with commonality. However, these models are only a simplification of the real world and several interactions and other relevant insights can be missed by the use of this tool.

The system dynamics model demonstrated that product commonality can have a deep impact on a product development project and that the negative effects of divergence are further amplified when larger divergence rates are experienced in the project. The divergence rates experienced in the project and used to calibrate the model above were based on actual measurements from a product development project, however, the magnitude and the behavior over time of the divergence rate may be dependent on various factors that are difficult to measure.

A relevant insight on the divergence rates was found in this case study; the divergence rate was found to be highly correlated with the original product commonality when the project started. Figure 5.18 and figure 5.19 show the dependence of divergence rates compared to their original commonality levels for both waves. The divergence rates in the second wave were measured using only two data points.

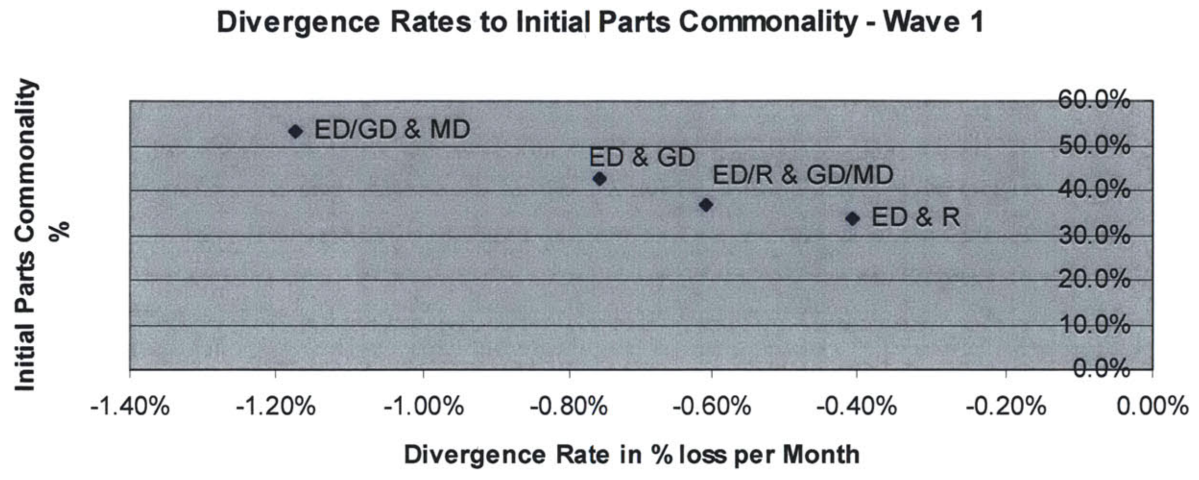


Figure 5.18: Divergence rate to original product commonality correlation for the first wave of vehicles.

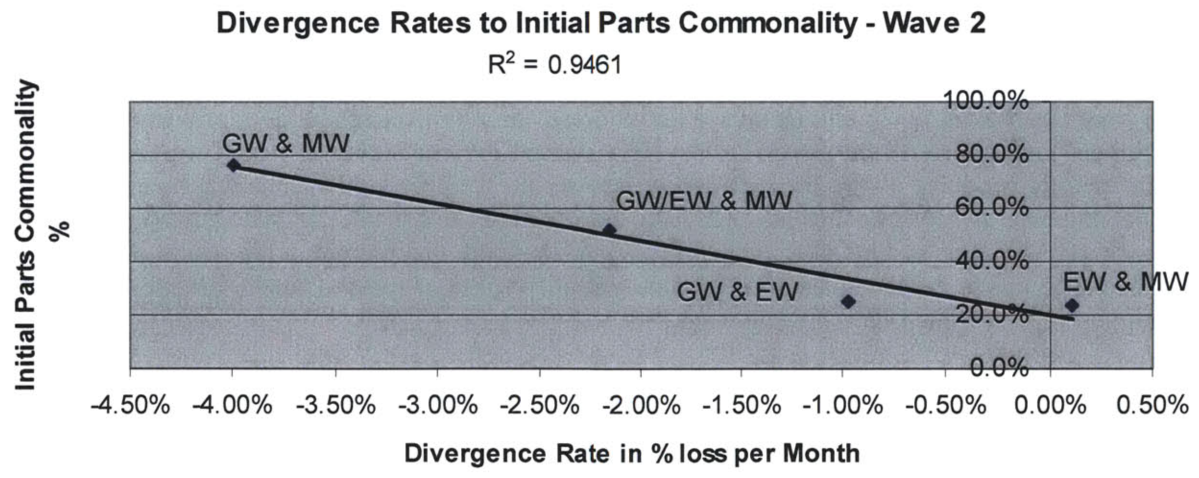


Figure 5.19: Divergence rate to original product commonality correlation for the second wave of vehicles.

One conclusion from the above figures is that there is a very high correlation ($R^2 > 0.9$) between the initial commonality and the divergence rate. When the vehicles share more parts, they are more likely to loose their commonality faster than the other vehicles, because they have more common parts to give up. The data suggests that common parts are an "unnatural" solution and the natural state of the system is toward unique designs. This was also experienced in various case study changes since commonizing parts portrays several difficulties and complex tradeoffs, while divergence allows the engineers to relax the tradeoffs and design more efficient solution for each of the products. This does not mean, however that the pursuit of commonality should be abandoned altogether. There appears to be an optimal level of commonality not only based on the well-accepted tradeoff between parts cost and individual derivative performance but also taking into account the divergence, lifecycle offset and productivity effects on project dynamics.

Another observation from these figures is the relative relevance of the projects inside the company. In both projects (Wave 1 and Wave 2), the Mayan vehicles were planned to share all their parts with the other two vehicles, and only minor changes were required due to local regulations and some unique market wants. The higher divergence rates were experienced for the Mayan projects, showing that follow on derivatives projects are also challenging to execute, even if most of the parts are shared. These difficulties in follow on projects have also been addressed in the literature (Boas, 2008; Lenfle et al, 2007).

The higher divergence rate is not the only challenge in follow-on derivative projects, but also the relative lack of attention to its development assuming that common parts will be developed for the lead product anyways. While an additional product may be a good opportunity for the product family overall, all the projects have their unique situations. Usually the integration resources are the ones more affected with the incremental products in the product family, as the complexity of adding new projects is usually assumed to be linear, however, it was found to be geometrical as detailed in the introduction in figure 1.6.

The resulting divergence rates experienced in the different projects were found to be naturally generated by the uncertainties in product development; however, several project decisions further decreased the original planned commonality, usually to create a better financial opportunity. The platform project manager has the ability to influence the design of the products to avoid excessive divergence and even to encourage convergence, but he/she also has the ability to influence the project behavior by adding or deleting products to the product family, an opportunity present only in these complex product families and experienced in the case study. In addition, the project manager has also additional tools to minimize the fundamental and the peripherial product variety by limiting the number of options available to the product which will further have an beneficial or a negative impact on divergence.

An additional finding from the case study is that product commonality is an information-intensive metric and is therefore rarely used to manage the project. The case study explained the accepted commonality metric, which was based on partial information of the different products. In general, product commonality was assumed to be much higher than the actual values measured and reported in the case study and used to develop the system dynamics models. Commonality was found to be highly influenced by the nature of the metric and the user of the information must understand how product commonality is measured if these metrics will be used to influence other product decisions. It should be noted, however, that for the same set of products commonality measures may differ depending on the level of decomposition of the BOM at which commonality is assessed.

In summary, the present chapter focused on the impact of divergence on product development projects using a system dynamics model to estimate this impact on the project schedule and to the project resources. The information from the case study suggested that product commonality was understood as a valuable tool to achieve efficiencies in product development; furthermore, a new project structure was created and a platform project manager was appointed to emphasize the relevance of product commonality. Nevertheless, divergence emerged as a normal characteristic of the development of complex product families. The information presented in this chapter concludes that actively managing product commonality is a valuable tool for the project manager to improve the individual projects performance and also for the overall product family project.

6. Conclusions and Future Research

6.1 Summary of Key Findings

The purpose of the present work was to extend the literature on product platforms by focusing on the dynamic aspects of the execution of complex product platform development projects, instead of the common approach in the literature that is focused only on the planning of product platforms. The central feature of this dynamic framework for platform projects product development is the fact that a fraction of product commonality is lost throughout the development of the products, a phenomenon called *divergence* (Boas, 2008). The present work can be considered an extension to the original Boas' research which was centered on the measurement and modeling of the behavior of divergence over time.

The present work proposed system dynamics as the suggested methodology to explain the different impacts of divergence on the development projects that are part of a product family. The system dynamics model was structured as two different projects A and B being developed concurrently, while sharing some of their parts and resources. The model was further divided into its two major activities: activities related to the design of the product and the activities related to the integration of those designs into a complete product.

The systems dynamics model was correlated with an extensive case study in the automotive industry (Chapter 3), which aimed at developing several new derivatives from a significantly updated product platform. The product platform under study was developed using a new project and organizational structure derived from the basic product architecture which differentiated the parts that were visible to the customer and usually unique to each derivative (Top Hat) to the parts that were not visible to the customer and which were expected to be reused (the platform) in order to achieve significant economies of scale with these components.

The set of conclusions in the following sub-sections represent the most relevant insights from this research. Most of the conclusions below are based on a single rich case study and the system dynamics model, and are not intended to create a new theory about product platforms and commonality but to further complement the existing knowledge of platform product development.

6.1.1. Platform Project Management

- A new term: "platform project management" was introduced to refer to the set of activities related to the development of a product platform and the coordination of different derivative projects.
- The key role of platform project management is ensuring that product commonality is achieved throughout its development by balancing and trading off all the designs among the entire product family and not making decisions for each product in isolation
- The greatest challenge of platform project management is to manage the complex tradeoffs of the platform "iron triangle" which is based on how the project and the product shares its design and its resources among numerous projects, instead of managing single projects in isolation.

6.1.2. Proposed New Project Structure based on Product Architecture

- A new project structure was proposed in the case study as an additional initiative to emphasize the relevance of product commonality. Each development project was divided into two projects: the "platform" project and the "Top Hat" project.

- The proposed structure was introduced to help the organization to manage the challenge of designing differentiated products, while maintaining high product commonality. The other main reason for the new structure was to spread the lead responsibility to several project managers to achieve better decisions and to reduce the workload for each.
- The new project structure proved to be an effective method to manage product commonality decisions and to spread the workload and key resource constraints better; however, the integration of the project was done in two steps, creating additional workload for the team members assigned to the integration of the overall product. The differentiated change control meetings proved to be the most relevant contribution to manage the visible and the non-visible portion of the product, and to balance the decisions with future products.
- The new project structure was not disruptive to the majority of the development organization. Specifically, the team members assigned to the design and development of the various components continued operating in the same manner.
- The project structure was enabled by the product development process that suggested that the vehicle should be developed in two workstreams: an upperbody and an underbody workstream to spread the risks of the development into two phases.

6.1.3. Product Commonality Measurement

- Two different methods of measuring product commonality were included as part of the case study. Commonality measurement is an information-intensive process that was performed in selected design freeze points and not used as part of the day to day project progress.
- The different methods for calculating commonality resulted in different perceptions of the efficiency of the platform development. The proposed corporate metric usually resulted in higher commonality, which was based on an incomplete view (the "control model") of the overall scope of the product platform.
- More frequent commonality assessments and a less information-intensive process would be preferred in order to actively manage product commonality and avoid non-beneficial divergence.
- The commonality metrics were derived from a predefined assembly level, which mostly created the need for unique part numbers; however, at lower levels of the BoM, several components were shared among derivatives. These "almost common" parts are called cousin parts and were found to be a relevant method to achieve the expected economies of scales without the need to reuse the exact same part number. Designing these components in a modular way allows to inexpensively creating a wide array of assemblies to comply with the various needs.

6.1.4. Qualitative Divergence Framework based on the Product Development Process

- Using the two different commonality assessment methods, divergence was present for all the different derivatives, concluding that divergence is a common phenomenon in complex product families. Divergence occurred in the product family under study even with a strong willingness, management support, and a new structure to enable product commonization.
- The original Boas' (2008) framework for potential sources and enablers of divergence was further complemented by investigating the qualitative effect of these sources and enablers on the different product development phases. Some of these sources and enablers were found in specific product development phases and others were found throughout the project.
- The lack of coordination as a divergence enabler was found to be caused by different processes and few personnel sharing between the first and the second wave of vehicles. The project hand-off from the planning to the executing team and the different tools related to these two phases were found to be a disruptive event that created significant coordination issues and negatively

impacted product commonality. Coordination is expected to be achieved within the project (cross functionally) and across projects.

- Four aspects were investigated as probable factors to improve overall product commonality. Using a design of experiments, it was concluded that a concurrent timing and similar product capacitbilities were the most relevant factors to influence higher product commonality over common markets or common manufacturing locations.
- Divergence can be beneficial or non beneficial, and the key role of the platform project manager is to make conscious product design decisions that improve the overall product family profitability, sometimes achieved by allowing divergence to happen. The key to effective decisions is the availability of quality information and accurate models to perform an adequate assessment for each change.

6.1.5. Lifecycle Offsets Findings

- Lifecycle offsets were found in the case study and proved to have a significant effect on product commonality. The lifecycle offsets were included in the original development plan to minimize the risk of the development, reduce the overall development costs by deferring some expenses, and to avoid a high peak demand for human resources .
- While the case study considered a lifecycle offset between the two waves, the concurrent development of the design of both vehicles, even if their design progress was unequal as they stood on different product development phases, was proven to be beneficial. The overlap between projects allowed the engineers to adapt their designs – with some noted exceptions - for all the different affected derivatives.

6.1.6. Design and Integration activities in Product Development

- The design and integration workstreams in product development were found to have unequal consequences derived from product commonality. The design workstreams were affected by both fundamental and peripherial variety, and the integration activities were found to be affected mostly by peripherial product variety.
- A simple model was derived to understand the sequential relationship of product design and product integration. The simple model was found to be a good example of the iron triangle framework, where scope, schedule and cost have to be traded off.
- Using the simple model, it was found that the overall project time can be improved more efficiently if more emphasis is given to the integration workstreams rather than the design workstreams.

6.1.7. Behavior of Divergence over Time –Divergence Rate

- Using the commonality data from the case study, a divergence rate was derived based on the % of lost commonality (or design uniqueness increase) for a specified period of time. The typical divergence rates ranged from 0.4% to 1.2% commonality loss per month.
- Commonality as it highest magnitude at the start of the progress and it decreases eventually throughout the development. The reason for this rather optimistic commonality assessment is the use of surrogate data and high level assessments. A more detailed assessment for product commonality would require having a full development team earlier in the development process. However, this may represent an unwanted project expense as there is a limit on the tasks that can be pulled ahead in a project. The design structure matrix (DSM) is a convenient tool to understand the nature of the tradeoff of what can be done earlier in the process.

- Product commonality was measured for all the different derivatives at three different dates, and the divergence rate was estimated using a linear regression of the commonality percentage. This same linear behavior was used as a key input to feed the system dynamics model.
- The divergence rate was found to be correlated with the initial product commonality. Products that share more parts initially are more likely to decrease commonality as common designs appear to be a high energy level situation and they eventually evolve to unique designs. Achieving and maintaining commonality over time requires significant coordination and willingness to trade off the different attributes.

6.1.8. Developed System Dynamics Model

- A system dynamics model was developed to reproduce the behavior of the projects included in the case study. The simulation included a concurrent simulation for two different projects which are sharing parts and resources. In addition, the model explicitly differentiated between the design workstream and the integration workstream which were related to each other.
- The system dynamics model included the design changes as the key feature of the design rework cycle. Using the case study, an average number of design changes was incorporated into the model to estimate the progress of the design and the integration workstream.
- Using the divergence rate and the original commonality assessment, the projects were found to be well correlated to the proposed systems dynamics model.

6.1.9. Causal Loop Diagrams

- Using the framework for product divergence sources and enablers, a platform project causal loop diagram was developed to understand the relevant feedback loops related to platform project management.
- The complete causal loop diagram concluded that divergence is not an external effect but a result of all the design changes, which are driven by the behavior of the team. Other factors like the economic incentives, the original level of commonality, and even the structure of the project and the team are also choices that can be controlled by the project stakeholders.
- The high number of changes is driven by numerous design optimization cycles. Every design change can have either a positive or negative effect on product cost and performance; however, all the changes inevitably increase the required investment for the product family, which was found to be the most difficult project objective to achieve.
- An infeasible iron triangle from the beginning of the project is the source of rework, further increased with an integral architecture of the project which allows change propagation to happen.
- All the different design decisions made throughout the project under study were categorized by the decision drivers and tradeoffs and were found to be correlated with the feedback loops found above to keep the project within its objectives. Project schedule was found to be the highest priority, followed by product quality and costs.
- Project schedule was found to be another relevant source of divergence which was not included in the original Boas' (2008) framework.
- Divergence was found to be a phenomenon that increases the scope of the project, rather than a feedback loop of the causal loop diagram. If the project scope is increased, additional resources or time will be required to achieve the desired project quality.

6.1.10. Impact of Divergence and Managerial Recommendations

- The system dynamics model was used to assess a hypothetical project which was intended to accurately capture the relevant inputs that affect divergence.
- Lifecycle offsets did not impact project performance directly; however, lifecycle offsets can be a relevant factor to amplify the lack of coordination or loss of experience (not modeled in this thesis) that could end in incremental divergence.
- Using the model, the effects of divergence were quantified and were found to be as high as a 22% schedule overrun or a 29% increase of the required personnel to achieve the planned project schedule. These effects grow proportionally to the divergence rate.
- Various countermeasures were investigated to keep the project schedule on time. All the managerial actions related to increasing the work rate were found to be effective to counteract the effects of divergence; however, they all have tradeoffs with cost or schedule.
- On the other hand, the countermeasures suggested to impact the rework cycle, either with less iterations or shorter iterations. These, however, were found to be ineffective since they are driven by the high design churn and because the result of divergence is an increased scope rather than a feedback effect.
- To keep all the product family projects within objectives, the best managerial recommendation is to avoid non-beneficial divergence rather than finding other solutions as a countermeasure for divergence. Therefore, actively managing product commonality is the most relevant recommendation to achieve a successful platform project.

6.2 Recommendations for Future Research

The conclusions above are based on a rich case study and a related system dynamics model aimed to understand the effects of divergence and lifecycle offsets for two projects in a product platform development. These findings can only be further validated with additional case studies and quantitative assessments of the actual effects of divergence in a product family. Besides including other industrial examples to validate these conclusions, the following key points represent some recommendations for future research.

- The lack of data availability pushed the system dynamics model to represent the behavior of divergence as a linear decrease over time. The actual divergence behavior should be measured in actual projects to understand this trend. Divergence should have an asymptotic (goal seeking) behavior towards a lower threshold of commonality. This threshold would represent an equilibrium point were forces towards more differentiation and towards more commonality are in balance with each other.
- The case study included only three product development stages; however, the behavior of divergence should be measured throughout the project for a complete project. Divergence should be expected also in the production ramp up phase of a project.
- The product architecture was found to have a significant effect on change propagation and therefore on divergence. It is suggested to further investigate the effects of divergence in modular and integral architectures.
- The case study emphasized the impact of full product commonality; however, the effects of divergence may be different if cousin parts are taken in to consideration. A more inclusive framework is required to understand the contribution of cousin parts to the product platform.

7. Bibliography

Alizon, F., Shooter, S.B. & Simpson, T.W., 2009. Assessing and improving commonality and diversity within a product family. *Research in Engineering Design*, 20(4), pp.241-253.

Allen, T.J., 1984. *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information Within the R&D Organization* 1st ed., Cambridge, Mass: MIT Press.

Automotive Industry Action Group., 1997. *Automotive project management guide.*, Southfield Mich.: Automotive Industry Action Group.

Boas, R.C. & Massachusetts, I.O.T., 2008. *Commonality in Complex Product Families: Implications of Divergence and Lifecycle Offsets*,

Books24x7, Inc.;Project Management Institute., 2008. *A guide to the Project management body of knowledge (Pmbok guide)*, Newtown Square, Pa. :: Project Management Institute,

Clark, K. & Fujimoto, T., 1991. *Product development performance : strategy, organization, and management in the world auto industry*, Boston Mass.: Harvard Business School Press.

Cusumano, M., 2010. Technology Strategy and Management: The Evolution of Platform Thinking. *Communications of the ACM*, 53(1), pp.32-34.

Cusumano, M.A., 1998. *Thinking Beyond Lean: How Multi-Project Management Is Transforming Product Development at Toyota and Other Companies*, New York: Free Press.

Dinsmore, P., 2006. *The AMA handbook of project management* 2nd ed., New York: AMACOM American Management Association.

Eckert, C., Clarkson, P.J. & Zanker, W., 2004. Change and customisation in complex engineering domains. *Research in Engineering Design*, 15(1), pp.1-21.

Eppinger, S.D. et al., 1994. A MODEL-BASED METHOD FOR ORGANIZING TASKS IN PRODUCT DEVELOPMENT. *Research in engineering design*, 6(1), pp.1-13.

Fixson, S.K., 2007. Modularity and commonality research: past developments and future opportunities. *Concurrent engineering, research and applications*, 15(2), pp.85-111.

Ford, D.N. & Sterman, J.D., 1998. Dynamic modeling of product development processes. *System Dynamics Review (Wiley)*, 14(1), pp.31-68.

Ford, H., 1923. *My Life and Work*, Garden City, N.Y: Doubleday, Page & co. Available at: http://www.gutenberg.org/ebooks/7213 [Accessed November 28, 2010].

Giffin M., de Weck O., Bounova G., Keller R., Eckert C., Clarkson P.J., "Change Propagation Analysis in Complex Technical Systems", *Journal of Mechanical Design*, 131 (8), 081010, August 2009

Holtta-Otto K., de Weck O.L., "Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints", *Concurrent Engineering*, Special Issue on Managing

Modularity and Commonality in Product and Process Development, 15 (2), 113-126, 2007

Kearney, A., 2010. *The 2010 Annual Automotive Study: Stars Aligned for a Robust Automotive Recovery?*, Southfield Mich.: A.T. Kearney. Available at: http://www.atkearney.com/index.php/Our-expertise/2010-annual-automotive-study.html [Accessed December 14, 2010].

Lenfle, S. et al., 2007. New Product development in a platform-driven organization: Towards platform lifecycle management. Available at: http://econpapers.repec.org/paper/haljournl/hal-00263329_5fv1.htm [Accessed November 17, 2010].

Lyneis, J.M., Cooper, K.G. & Els, S.A., 2001. Strategic management of complex projects: a case study using system dynamics. *System Dynamics Review (Wiley)*, 17(3), pp.237-260.

Lyneis, J.M. & Ford, D.N., 2007. System dynamics applied to project management: a survey, assessment, and directions for future research. *System Dynamics Review (Wiley)*, 23(2/3), pp.157-189.

Macduffie, J.P. & International, M.V.P., 1993. *Product Variety and Manufacturing Performance: Evidence from the International Automotive Assembly Plant Study*, Cambridge, Mass.: International Motor Vehicle Program, Massachusetts Institute of Technology.

Meyer, M.H., 1992. *The Product Family and the Dynamics of Core Capability*, Cambridge, MA: International Center for Research on the Management of Technology, Sloan School of Management, Massachusetts Institute of Technology.

Meyer, M.H. & Lehnerd, A.P., 1997. *The Power of Product Platforms: Building Value and Cost Leadership*, New York: Free Press.

Muffatto, M., 1999. Platform strategies in international new product development. *International journal of operations & production management*, 19(5/6), p.449.

Muffatto, M. & Roveda, M., 2000. Developing product platforms:: analysis of the development process. *Technovation*, 20(11), pp.617-630.

Pine, B.J., 1993. *Mass Customization: The New Frontier in Business Competition*, Boston, Mass: Harvard Business School Press.

del Puerto Valdez, A. & Massachusetts Institute of Technology.;System Design and Management Program., 2010. *Analysis of the product development process for geographically distant teams in vehicle tophat design*,

Ramdas, K., 2003. Managing Product Variety: An integrative review and Research Directions. *Production and Operations Management.*, 12(1), p.23.

Repenning, N.P., 2000. A dynamic model of resource allocation in multi-project research and development systems. *System Dynamics Review (Wiley)*, 16(3), pp.173-212.

Robertson, D. & Ulrich, K., 1998. Platform Product Development. *Sloan Management Review*, p.30.

Sanderson, S. & Uzumeri, M., 1995. Managing product families: The case of the Sony Walkman.

Research Policy, 24(5), pp.761-782.

Seepersad, C., Hernandez, G. & Allen, J., 2000. A quantitative approach to determining product platform extent. In *Design Engineering Technical Conference. Proceedings of the*. ASME Design Engineering Technical Conferences. Baltimore, MD: American Society of Mechanical Engineers, p. 1073. Available at: http://library.mit.edu/F/NYGGHN9IRN8YASFJPVCI5V1YJG7BD62MUD1BNK9M3KVPCTXNG3-30242?func=find-acc&acc_sequence=017955086.

Siddiqi A., Bounova G., de Weck O., Keller R., Robinson B., "A posteriori Design Change Analysis for Complex Engineering Projects", *Journal of Mechanical Design*, MD-10-1232, April 2011 (in press)

Mahmoud-Jouini, Sihem Ben & Lenfle, Sylvain 2010. Platform re-use lessons from the automotive industry. *International Journal of Operations & Production Management*, 30(1), p.98.

Simpson, T.W., Siddique, Z., Jiao, J. 2006. *Product Platform and Product Family Design: Methods and Applications*, New York: Springer Science+Business Media.

Sterman, J., 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*, Boston: Irwin/McGraw-Hill.

Stewart, B., 2008. Separated at Birth. *Popular Mechanics*, 185(5), pp.80-85.

Suh, E.S., de Weck O.L., Kim I.Y. and Chang D., "Flexible platform component design under uncertainty", *Journal of Intelligent Manufacturing*, Special Issue on Product Family Design and Development, 18 (1), 115-126, February 2007

Suh E.S., de Weck O.L., and Chang D., "Flexible product platforms: framework and case study", *Research in Engineering Design*, 18 (2), 67-89, 2007

Thevenot, H.J. & Simpson, T.W., 2007. A comprehensive metric for evaluating component commonality in a product family. *Journal of Engineering Design*, 18(6), pp.577-598.

Thevenot, H.J. & Simpson, T.W., 2006. Commonality indices for product family design: a detailed comparison. *Journal of Engineering Design*, 17(2), pp.99-119.

Ulrich, K., 1995. The role of product architecture in the manufacturing firm. *Research Policy*, 24(3), pp.419-440.

Ulrich, K.T., 2008. *Product Design and Development* 4th ed., Boston: McGraw-Hill Higher Education.

de Weck, O. & Lyneis, J., 2010. *Successfully Designing and Managing Complex Projects* First Edition (Draft)., MIT Press.

de Weck, O., "Determining Product Platform Extent", Chapter 12, *Product Platform and Product Family Design: Methods and Applications*, T. W. Simpson, Z. Siddique and J. Jiao, eds., Springer, New York, pp. 241-301, 2005