Proceedings of DETC2000
2000 ASME Design Engineering Technical Conferences
September 10 - 13, 2000, Baltimore, MD

DETC2000 / DFM-14021

DESIGN FOR VARIETY: A METHODOLOGY FOR DEVELOPING PRODUCT PLATFORM ARCHITECTURES

Mark V. Martin
Kosuke Ishii
Department of Mechanical Engineering
Stanford University
Stanford, CA
mvmartin@stanford.edu

ABSTRACT

Developing a robust, product platform architecture brings an important competitive advantage to a company. The major benefit is reduced design effort and time-to-market for future generations of the product. This paper describes a step-by-step method that aids companies in developing such a product platform architecture. Using the concept of specification "flows" within a product development project, the design for variety (DFV) method develops two indices to measure a product's architecture. The first index is the Generational Variety Index (GVI), a measure for the amount of redesign effort required for future designs of the product. The second index is the Coupling Index (CI), a measure of the coupling among the product components. The design team uses these two indices to develop a decoupled architecture that requires less design effort for follow-on products.

KEYWORDS: Architecture, Product platform & family, Modularity, Information Flow, Coupling

1. INTRODUCTION

Related literature

Design for Variety (DFV) is a series of structured methodologies to help design teams reduce the impact of variety on the life-cycle costs of a product (Martin and Ishii 1996; Martin and Ishii 1997). Various authors have explored issues dealing with the strategic benefits of developing product platforms and the management of families.

Pine (1993) discusses the need for product variety in today's marketplace. Sanderson & Uzumeri (1995) use a case study on Sony Walkmans to show how understanding the market, the use of "strong" design, the effective division of labor, and manufacturing flexibility aid in rapid model development. Sanderson (1991) considers how design management strategy can affect design costs.

Robertson & Ulrich (1998) discuss planning for product platforms. They encourage the use of platform development early on and state that it must include consideration of marketing, design, and manufacturing issues.

Galsworth (1994) describes the Variety Effectiveness Program (VEP) – a methodology for helping companies decrease the complexity of variety. She uses six analysis tools in VEP to help guide companies. These cover the areas of 1) unique vs. shared parts, 2) modularity, 3) reduction of part count, 4) design for assembly, 5) range of component specifications, and 6) trends in product and component specifications.

Fujita et al (1998) use optimization techniques to estimate the best architecture for a family of aircraft. Adler et al (1995; 1996) consider design as a stochastic processing network with engineering resources as workstations and projects as jobs that flow between the workstations. Their process model provides a useful framework for understanding bottlenecks in designs and how changes might be made to reduce the bottlenecks.

Erens (1996) characterizes development under functional, technology, and physical domains. He uses this characterization to help develop product platforms. Gonzalez-Zugasti et al (1998) use a metamodel of the technical performance requirements and costs to optimize the design of a family of spacecraft based on a common platform.

Tseng & Jiao (1998) develop the product family architecture (PFA) model to handle the tradeoffs between diversity of customer requirements and reusability of design and process capabilities.

The literature review showed a growing interest in the area of product platform architectures. It also showed an opportunity for a more detailed and prescriptive approach to developing product platforms. From this came the research presented in this paper. It gives a detailed, step-by-step approach to help a design team develop a product platform.

Definition of architecture

Ulrich (1995) refers to product architecture as the “scheme by which the function of a product is allocated to physical components.” He defines it more precisely as: 1) the arrangement of functional elements; 2) the mapping from functional elements to physical components; 3) the specification of the interfaces among interacting physical components.

By definition, any product design meets all three of Ulrich’s requirements for architecture. A design must have an arrangement of functional elements, a mapping between function and structure, and specified interactions among components. Thus, any design for a single product has an architecture.

A product family can also have an architecture. A family architecture implies that the different products have a common arrangement of elements, common mapping between function and structure, and common interactions among components. A product family architecture only exists if there is this commonality.

Our method seeks a structured method that aids in developing 1) the arrangement of functional elements, 2) the function-structure mapping, and 3) the interface specifications for a product family. In essence, the DFV method gives operational detail to Ulrich’s architecture concept.

What is the goal of developing an architecture?

The purpose for developing an architecture for a product line is to maximize the profit potential for the company. Our work seeks to aid engineers in creating designs that leverage current design effort across future products and thus reduce development costs. These products will have an architecture that requires minimal changes to meet future marketplace needs. Meeting these future needs would be relatively simple in a deterministic world. However, the uncertainty in future customer needs, technology changes, competitor response, etc. complicates the planning of a product that can leverage current design efforts.

To develop a method that helps leverage the design effort, we first look at the factors external to the company that will cause a design to change over time. By understanding these “drivers” of change, we can begin to plan the product line such that it isolates components that are likely to change. This understanding will help minimize design effort for future products and commonize design structures across generations.

Section 2 discusses the Generational Variety Index. Section 3 covers the Coupling Index. Section 4 discusses how these indices are applied in the Design for Variety (DFV) method.

Product variety: Spatial & generational

Our work deals with two types of variety to consider when developing the architecture of a product: 1) variety within the current product line being designed, and 2) variety across future generations of the product. We refer to the variety in the current product being designed as "spatial" variety. The variety across generations is referred to as "generational" variety. These terms are illustrated in Figure 1.

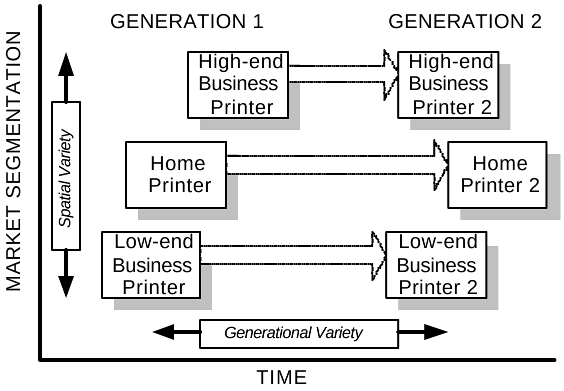

Figure 1: Spatial and Generational Variety

This paper focuses on developing a design that can be easily leveraged for future products. However, the concepts can also be applied to spatial variety considerations.

2. GENERATIONAL VARIETY INDEX (GVI)

The Generational Variety Index (GVI) is an indicator of which components are likely to change over time. The GVI is defined as:

> *The Generational Variety Index (GVI) is an indicator of the amount of redesign required for a component to meet the future market requirements.*

After reading this section the reader should be able to understand and duplicate the Generational Variety Index.

Drivers of generational changes

The GVI is based on an estimate of the required changes in a component due to external (i.e., non-controllable) factors. Examples of such external drivers are customer needs, reliability requirements, reduced costs, etc. A more detailed listing is shown in Table 1.

Table 1: External drivers of generational change

Customer Requirements Changing performance needs (including size, style, weight, etc.) New environmental constraints (temperature, humidity, vibration, etc.) New functions (due to new markets or new enabling technologies) Reliability improvements Etc.
Cost Reduction Reduce amount of material Change material type Remove redundant components Reduce assembly time Use lower cost technology Reduce serviceability requirements Reduce serviceability time Improve component manufacturing process Etc.
Regulations, Standards, etc. Changing government/industry regulations or standards Competitor introduction of improved product (higher quality or lower price) Obsolescence of parts Etc.

The changes in these external drivers can cause changes in the components over time. In this paper the external drivers are measured in the form of Engineering Metrics (EM's).

Estimating the GVI is the first step, followed by the generation of the Coupling Index (described in the next section). These two indices are then used in the DFV method. This method aids the team in developing an architecture that can be easily leveraged. To demonstrate the DFV process, a simplified inkjet printer example will be used throughout this document. Only a few subsystems (referred to as "components" from here forward) of the printer will be considered.

The five components considered for the printer are the PCA / Firmware, Print Cartridge, Carriage Sub-System, Input/Output Tray, and Feed Sub-System. An exploded view of the printer is shown in Figure 2 (courtesy of Hewlett-Packard, 1996). The components are marked on the drawing.

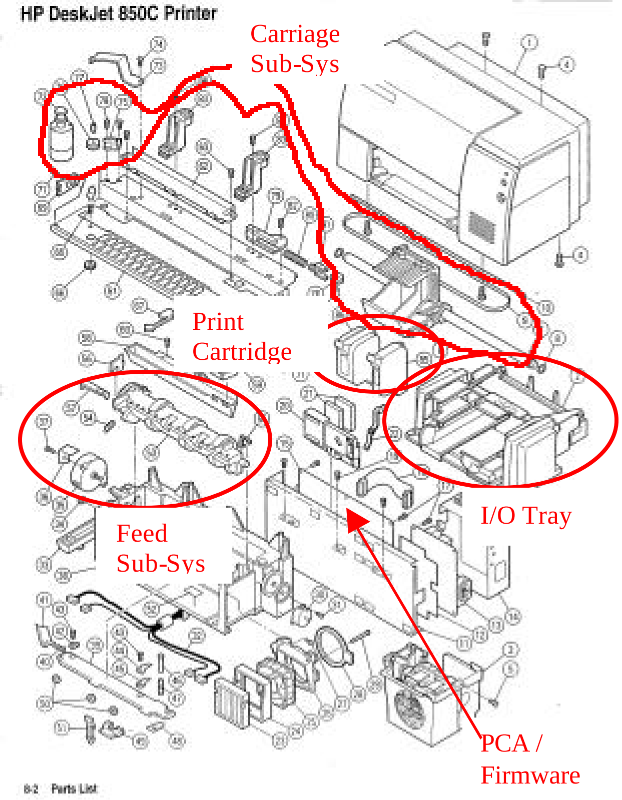

Figure 2: Exploded view of inkjet printer indicating component sub-systems

Using Quality Function Deployment (QFD) for input

To generate the GVI, the team must first estimate what external drivers might require the product to change over time. Note that the time period considered is based on how long the team wishes the architecture to last.

To generate the GVI we use a modified Quality Function Deployment (Hauser and Clausing 1988) structure. For this simplified printer example, we consider customer requirements and cost to be the drivers of change. Changing regulations, standards, etc. could also be added to the matrix if desired.

QFD Phase I

QFD Phase I lists the customer needs and their relationship to engineering metrics. In the inkjet printer example, a subset of the customer requirements is listed.

Items such as "prints fast", "good image quality", "low noise", and "compact" are a few examples of the customer needs for this product. The engineering metrics for the various needs are measurable items such as "pages per minute (PPM)", "dots per inch (DPI)", "decibels," and "footprint". These are a translation of the subjective customer requirements into quantifiable engineering specifications (Figure 3).

	Engineering Metrics (EM)					
Customer Requirements	Pages per minute (PPM)	Dots per inch (DPI)	Decibels (dB)	Footprint (sq in)	MTBF (hrs)	Unit cost ($)
Prints fast	X					
Good image quality		X				
Low noise			X			
Compact				X		
Reliable					X	
Low cost						X

Figure 3: GVI QFD Phase I

QFD Phase II

QFD Phase II maps the engineering metrics from Phase 1 to the components used in the design. The mapping for the printer example is shown in Figure 4. An "X" indicates that the component can affect the engineering metric.

	Components				
Engineering Metrics	PCA/Firmware	Print Cartridge	Carriage Motor Assembly	Input/Output Paper Tray	Feed Motor Assembly
Pages per minute (PPM)	X	X	X	X	X
Dots per inch (DPI)	X	X	X		X
Decibels (dB)		X	X	X	X
Footprint (sq in)	X		X	X	
Reliability (MTBF)	X	X	X	X	X
Unit Cost ($)	X	X	X	X	X

Figure 4: QFD Phase II

Since cost, reliability, and standards (not shown in this printer example) are also external drivers for any product, these should be added to the engineering metrics if they are not already included. In this example, the PCA/Firmware, Print Cartridge, and Feed Motor Sub-System all have a impact on the PPM metric. The PCA/Firmware also has an impact on the DPI, Footprint, Reliability, and Unit Cost.

GVI steps

The mapping of QFD Phase II helps in developing the GVI. The purpose of the GVI is to estimate how much component redesign effort is required to meet the future engineering metrics. The GVI number will be different for different architectures.

A number of different approaches for determining the GVI were considered and tried. The goal was that it be easy to understand and to use. In the end, direct input from the team members was determined the best process. The method for determining the GVI is described below.

GVI Step 1: Determine market & desired life of product platform

An understanding of where the market is headed is critical to the DFV method. Also, the team must determine how long they would like the product platform to last. For the printer example, the period is two years and four different products are envisioned. Methods to help map the future product plans are discussed by Wheelwright and Sasser (1989) and Wheelwright and Clark (1992). The markets this printer platform is attempting to satisfy are shown in Table 2.

Table 2: Markets and introduction dates

Market	Description	Introduction Date
	Development Start	*Jun-99*
Current	Home	Dec-99
Future 1	Business (Low Volume)	Jun-00
Future 2	Home (Lower Cost)	Feb-01
Future 3	Business (Improved Perf.)	Nov-01

GVI Step 2: Create QFD matrix

If not already available, create a simplified Phase I and Phase II QFD. See Figure 3 and Figure 4 for an example.

GVI Step 3: List expected changes in customer requirements

Add a column to Phase I estimating qualitatively (High/Medium/Low) the range of change for the customer requirements (see Figure 5). This is a simple step to get the development team to think about how the customer needs are changing. “High” indicates that this is a rapidly changing customer need and that large changes in it will be required.

	Engineering Metrics (EM)						
Customer Requirements	Pages per minute (PPM)	Dots per inch (DPI)	Decibels (dB)	Footprint (sq in)	MTBF (hrs)	Unit cost ($)	Expected range of change over next 2 years
Prints fast	X						H
Good image quality		X					M
Low noise			X				L
Compact				X			L
Reliable					X		M
Low cost						X	H

Figure 5: QFD Phase I with expected changes in customer requirements

GVI Step 4: Estimate engineering metric target values

This step is to determine the engineering metric target values (EMTV) for the period the product platform is being developed. The target values could be based on information from conjoint analysis, trend analysis, expected new markets, expected competitor introduction of products, etc. For this example, the estimated target values are estimated based on previous trends and marketing data. More formal methods, such as Yu et al (1998), give a more detailed approach to estimating future target values. The estimated future values for the printer are shown in Figure 6.

	Engineering Metrics (EM)						
Customer Requirements	Pages per minute (PPM)	Dots per inch (DPI)	Decibels (dB)	Footprint (sq in)	MTBF (hrs)	Unit cost ($)	Expected range of change over next 2 years
Prints fast	X						H
Good image quality		X					M
Low noise			X				L
Compact				X			L
Reliable					X		M
Low cost						X	H
EM Target Values (EMTV)							
Current Market	6	600	42	216	1500	100	
Future Market 1	8	600	48	216	2000	125	
Future Market 2	6	600	39	216	1500	75	
Future Market 3	12	1200	42	150	2500	100	

Figure 6: QFD Phase I with EM target values added

GVI Step 5: Calculate normalized target value matrix

This information is used to graphically display the changes for the target values. This step is skipped for this shortened DFV description.

GVI Step 6: Create GVI matrix

To determine the GVI matrix the team uses its engineering expertise and judgment to estimate the cost of changing the component to meet the most stringent future EM target values. The GVI matrix uses a 9/6/3/1 rating system for these estimates. For each EM/Component node in the matrix, the team estimates the component redesign costs (including design effort, tooling, and testing) required to meet the future target value for that engineering metric. These costs are expressed as a percentage of the original cost to design.

Table 3: GVI matrix rating system

Rating	Description
9	Requires major redesign of the component (>50% of initial redesign costs)
6	Requires partial redesign of component (<50%)
3	Requires numerous, simple changes (<30%)
1	Requires few, minor changes (<15%)
0	No changes required

For instance, the EM for print speed starts at 6 PPM and ranges up to 12 PPM. The team would decide if the PCA / Firmware, the Print Cartridge, Carriage Sub-System, I/O Paper Tray, or Feed Sub-System would require a major redesign, partial redesign, etc. This is based on the engineering expertise and judgment of the team. The QFD

Phase II matrix is helpful because it gives an indication of how important each component is to that particular EM. An example of the Phase II matrix with GVI input is shown in Figure 7.

	Components				
Engineering Metrics	PCA/Firmware	Print Cartridge	Carriage Sub-system	Input/Output Paper Tray	Feed Sub-System
Pages per minute (PPM)	6	9	3		3
Dots per inch (DPI)	6	9	3		1
Decibels (dB)		1	3		1
Footprint (sq in)	1		1	3	1
Reliability (MTBF)	3	3	1	1	1
Unit cost ($)	3	3	1	1	1

Figure 7: Phase II matrix with GVI input

GVI Step 7: Calculate GVI

The GVI for each component is calculated by summing each of the columns of the GVI matrix. The GVI calculation is shown in Figure 8. The application of measure theory concepts (Krantz and Suppes 1971) to the GVI shows that it maintains ordinal and ratio relationships.

	Components				
Engineering Metrics	PCA/Firmware	Print Cartridge	Carriage Sub-system	Input/Output Paper Tray	Feed Sub-System
Pages per minute (PPM)	6	9	3		3
Dots per inch (DPI)	6	9	3		1
Decibels (dB)		1	3		1
Footprint (sq in)	1		1	3	1
Reliability (MTBF)	3	3	1	1	1
Unit cost ($)	3	3	1	1	1
GVI	19	25	12	5	8

Figure 8: GVI calculation

These EM to Component linkages and weightings for the printer are illustrated in Figure 9.

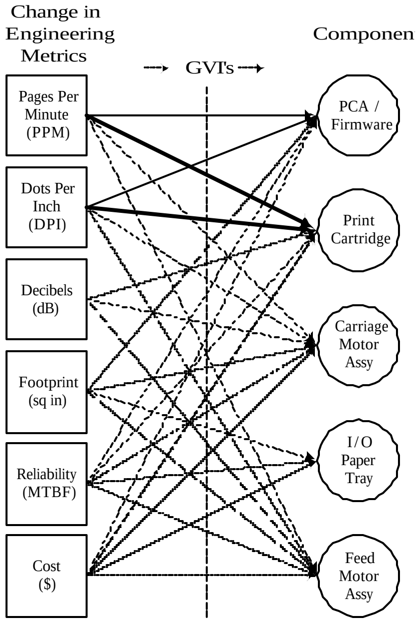

Figure 9: Illustration of linkages between EM and components

3. COUPLING INDEX (CI)

As discussed in previous sections, there are various external drivers for changing a design. The changes created by these drivers may in turn require other changes within the design. Such second-order changes do not directly enhance the value of the product, except to the extent that they support the first-order changes. These second-order (and higher) effects are created by the interaction, or "coupling", within the design. It became quickly apparent that understanding coupling within a design was crucial for developing architectures robust to future changes in customer requirements. The definition of coupling (Ulrich 1995) used in this paper is shown below.

> *Two components are considered coupled if a change made to one of the components can require the other component to change.*

This section develops our coupling index (CI) which is defined as:

> *The Coupling Index indicates the strength of coupling between the components in a product. The stronger the coupling between components, the more likely a change in one will require a change in the other.*

The CI is a measure of the first-order coupling between the components. The section begins with a detailed introduction to the definition of coupling as well as related work on this topic. This information is necessary to fully understand and use the DFV method.

Developing the Coupling Index is approached by considering the "specification flows" among components. These specification flows are defined as the design information that must be passed between designers to design their respective components. By mapping out the specification flows early in the design process, the team explicitly describes the relationships that couple the parts. Figure 10 shows the process for calculating the Coupling Index.

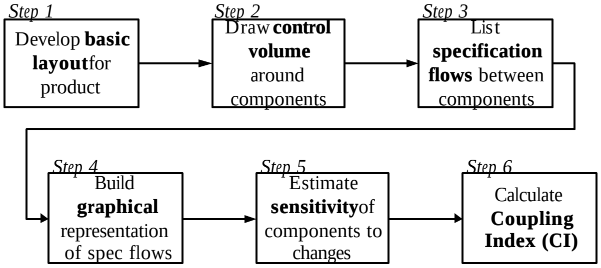

Figure 10: Flow chart of coupling index development

CI Step 1: Develop basic physical layout for the product

In order to generate the coupling index for a product, the basic technology to be used and the general layout of the product must be known. Without this, it would be difficult to determine how subsystems, subassemblies, or parts are coupled.

For a printer, the coupling within an inkjet approach and a laserjet approach is different, so specifying the technology to be used is necessary to determine the coupling index. Even specifying the technology would require more detail about the device. Within the inkjet architecture, different approaches could be used to apply the ink, move the paper, hold the paper, etc.

Once this basic information is determined, the coupling index can being to be developed. As more detail is brought to the design, the coupling index will evolve as new linkages between components are added and deleted.

CI Step 2: Draw control volume around components

A control volume (CV) is a boundary around a system indicating the flows into and out of that system. For the DFV method, the control volumes are "drawn" around each component. If possible, these control volumes should be approximately at the same level of complexity (i.e., do not list a "screw" as one component and a "power supply" as another).

CI Step 3: List specification flows required between components

For each control volume, have the engineer(s) list the specifications they need to *receive* about each of the other control volumes. Have the engineers also list the specifications that they expect to *supply* to each of the other control volumes. Do not assume any precedence among the components during this stage.

Reconcile the differences between the expected specification flows. Put these specification flows between components into matrix form. The top row of the matrix lists the components *supplying* the information; the left column lists the components *requiring* the information. Figure 11 shows a portion of specifications for the printer..

	Components supplying specifications			
Components receiving specifications	PCA / Firmware	Sensitivity	Print Cartridge	Sensitivity
PCA / Firmware			Resistance	9
			# of nozzles	3
			Nozzle pitch	3
			# of inks	3
			Firing rate	3
			Ink viscosity	3
			Drying time	3
Print Cartridge	Voltage	6		
	Firmware	1		

Figure 11: Partial CI matrix of specification flows

Figure 11 shows the specification flows between the Print Cartridge and the Printer Circuit Assembly / Firmware control volumes. In this example, the team has determined that changes in the "resistance", "# of nozzles", "drying time", etc. for the Print Cartridge can cause a change to the PCA / Firmware.

CI Step 4: Build a graphical representation of the specification flows

The next step is to build a graphical representation of the flows. This is optional but is useful in visualizing the flows between components (Figure 12).

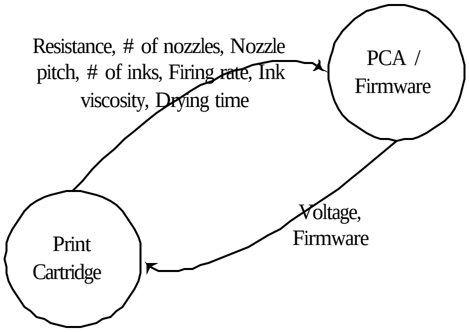

Figure 12: Graphical representation of specification flows

This listing of supplied and required information is useful for both spatial and generational variety. Those components that supply numerous specifications to other components are items that the design team would like to remain static in order to minimize redesign effort. The next steps for concept phase coupling are to determine a quantification index for the specification flows.

CI Step 5: Estimate sensitivity of components to changes

For each specification, the team estimates the sensitivity of each component to a small change in that specification. If a small change in the specification requires a change in the component, then the component has a High sensitivity. If the specification requires a large change to create a change in the receiving component, then it has a Low sensitivity. The High sensitivity specifications are given a rating of 9. The Low sensitivity specifications are given a rating of 1. Table 4 lists the descriptions of the numerical ratings.

Table 4: CI rating system for sensitivity of specifications

Rating	Description
9	Small change in specification impacts the receiving component (High Sensitivity)
6	Medium High Sensitivity
3	Medium Low Sensitivity
1	Large change in specification impacts the receiving component (Low Sensitivity)
0	No specifications affecting component

For this rating system, it is assumed that the "impact" caused by a specification change is equivalent and linear across all components. Figure 13 shows the sensitivity rating applied to a portion of the printer example.

	Components supplying specifications					
Components receiving specifications	PCA / Firmware	Sensitivity	Print Cartridge	Sensitivity	Carriage Sub-Assembly	Sensitivity
PCA / Firmware			Resistance # of nozzles Nozzle pitch # of inks Firing rate Ink viscosity Drying time	9 3 3 3 3 3 3	Motor speed Resistance Torque	3 6 1
Print Cartridge	Voltage Firmware	6 1			X dimension Y dimension Z dimension	6 6 6
Carriage Sub-Assembly	Voltage Firmware	6 1	Weight X dimension Y dimension Z dimension	3 6 6 6		

Figure 13: Partial CI matrix of specification flows including sensitivity ratings

For example, even a small change in the nominal resistance of the print nozzle heaters requires a change in the PCA / Firmware since the power input to the print nozzles is a critical in defining how the ink will be ejected from the nozzle. Because of this, the resistance of the Print Cartridge (PC) has a High (9) sensitivity. The same is true for the voltage since even a small change will require a change in the Print Cartridge resistance.

CI Step 6: Calculate coupling index

From the coupling matrix, two indices are derived. The sum for a column indicates the strength of the information supplied by that component to other components and is referred to as the Coupling Index – Supply (CI-S). The sum for a row is information being received by each component and is referred to as the Coupling Index – Receive (CI-R). The definition of these indices is shown below.

*Coupling Index – Receiving (**CI-R**): The CI-R indicates the strength of the specifications that a component **receives** from other components.*

*Coupling Index – Supplying (**CI-S**): The CI-S indicates the strength of the specifications that a component **supplies** to other components.*

For each column and row, sum the sensitivities (shown in Figure 14). For example, the CI-S for the

Print Cartridge is 48, which means that its design has a strong impact on other components in the design. The PCA/Firmware has a relatively high CI-R indicating the other components have a strong impact on it.

	Components supplying specifications						
Components receiving specifications	PCA / Firmware	Sensitivity	Print Cartridge	Sensitivity	Carriage Sub-Assembly	Sensitivity	**CI-R**
PCA / Firmware			Resistance # of nozzles Nozzle pitch # of inks Firing rate Ink viscosity Drying time	9 3 3 3 3 3 3	Motor speed Resistance Torque	3 6 1	**37**
Print Cartridge	Voltage Firmware	6 1			X dimension Y dimension Z dimension	6 6 6	**25**
Carriage Sub-Assembly	Voltage Firmware	6 1	Weight X dimension Y dimension Z dimension	3 6 6 6			**28**
CI-S		**14**		**48**		**28**	

Figure 14: Partial CI matrix including CI's

The CI-S and CI-R indicate how tightly coupled a component is. A high CI-S indicates that the component supplies a lot of necessary information to other components. If that component is changed, it has a higher likelihood of causing changes in other components. A high CI-R for a component indicates a higher likelihood it will require changes due to other components being changed. A graphical representation and full coupling index results for the printer example are shown in Figure 15 and Figure 16.

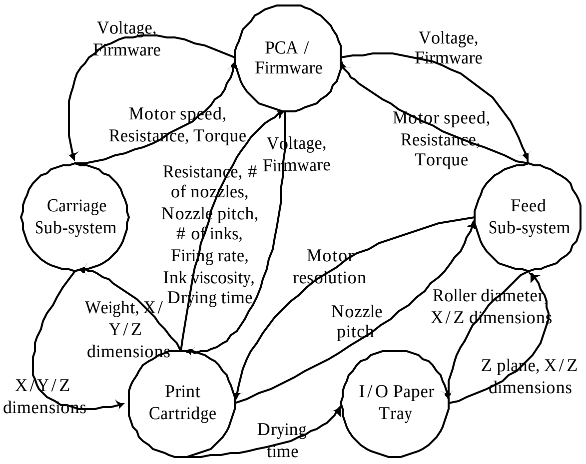

Figure 15: Graphical representation of specification flows

	Components supplying specifications										
Components receiving specifications	PCA / Firmware	Sensitivity	Print Cartridge	Sensitivity	Carriage Sub-Assbly	Sensitivity	I/O Paper Tray	Sensitivity	Feed Sub-Assembly	Sensitivity	**CI-R**
PCA / Firmware			Resistance # of nozzles pitch # of inks Firing rate Ink viscosity Drying time	9 3 3 3 3 3 3	speed Resistance Torque	3 6 1			Motor speed Resistance Torque	3 6 1	**47**
Print Cartridge	Voltage Firmware	6 1			dimension dimension dimension	6 6 6			resolution	9	**34**
Carriage Sub-Assembly	Voltage Firmware	6 1	Weight dimension dimension dimension	3 6 6 6							**28**
Input / Output Paper Tray			Drying time	3					Roller diameter X dimension Z dimension	9 3 3	**18**
Feed Sub-Assembly	Voltage Firmware	6 1	pitch	6			Z plane dimension dimension	6 3 3			**25**
CI - S	**21**		**57**		**28**		**12**		**34**		**152**

Figure 16: Complete CI matrix including CI's

As shown in Figure 16, the Print Cartridge has the largest CI-S. This indicates that is more tightly coupled within the design than the other components. A redesign of this component has a strong potential for requiring changes in other components. The drivers of this large CI-S are 1) the Print Cartridge supplies lots of specifications to other component, and 2) many of the components requiring these specifications are sensitive to any changes.

4. DESIGN FOR VARIETY (DFV) METHOD

The development of the Generational Variety Index (GVI) and Coupling Index (CI) is an important process. It gives the project team a more explicit understanding of the external drivers of change and of how changes may propagate throughout the design.

This section describes how these indices are used to develop a product platform architecture that is more robust to changes from the external environment. The generation of the indices and their application to architecture development constitutes the Design for Variety (DFV) method. The method uses the indices to focus on the most critical areas in developing the architecture.

The previous two sections looked at the reasons a component will change over time. As discussed, there are two causes for a component to change – external drivers (measured by the GVI) and internal drivers (measured by the CI-R). The external drivers come from areas outside the design team's control (changing customer requirements, regulations, competitor introductions, etc.). The internal drivers come from the coupling between the product components. These drivers are illustrated in Figure 17.

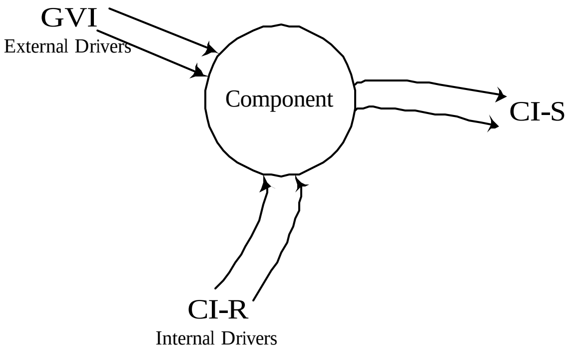

Figure 17: Illustration of drivers of component change

The complete specification flows for this simplified printer example are shown in Figure 18.

The DFV Method captures these flows and their strengths to guide the team in developing the architecture. To accomplish this, the method uses two heuristics for helping the team determine the critical components affecting the design effort. The team then uses three different approaches to modify the architecture to reduce these effects. These are described in the following sections.

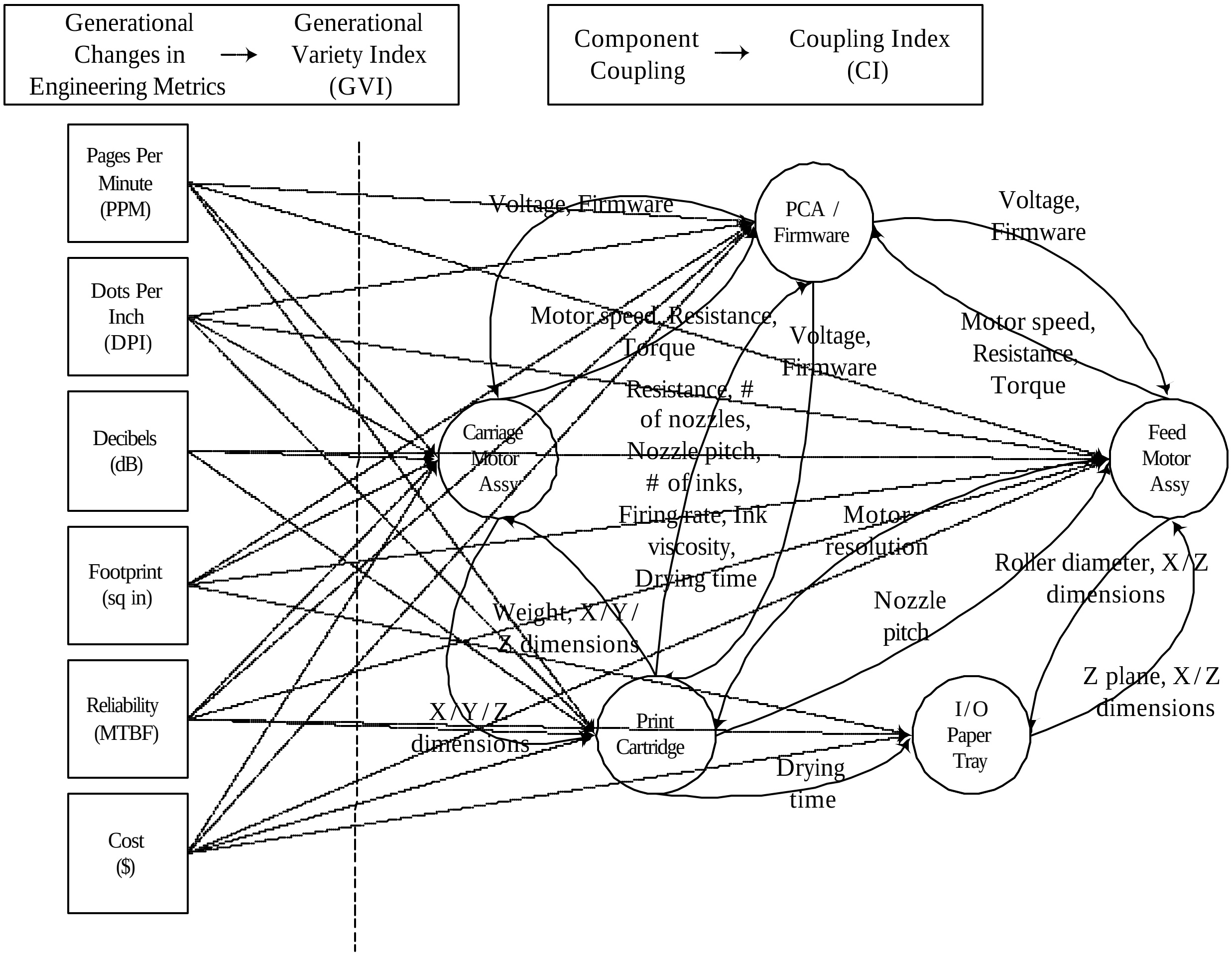

Figure 18: GVI & CI specification flows

DFV method steps

DFV Step 1: Generate GVI and CI for the design

As described in previous sections, the Generational Variety Index and Coupling Indices are generated for the product.

DFV Step 2: Order the components

a) Rank Order the GVI

Based on the GVI, rank order the components from highest to lowest. These are the components that are most likely to change over the product platform time period due to external drivers. The results for the printer are shown in Table 5.

Table 5: Rank ordering of GVI

Component	GVI
Print Cartridge	25
PCA / Firmware	19
Carriage Sub-System	12
Feed Sub-System	8
Input / Output Paper Tray	5

b) Include Coupling Indices and design costs

Add Coupling Indices for each of the components as shown in Table 6.

Table 6: GVI, CI's, & design costs

Component	GVI	CI-R	CI-S	Total Design Costs
Print Cartridge	25	34	57	$PC
PCA / Firmware	19	47	21	$PCA
Carriage Sub-System	12	28	28	$CSS
Feed Sub-System	8	28	34	$FSS
Input / Output Paper Tray	5	18	12	$IO

DFV Step 3: Determine where to focus efforts -- where to standardize or modularize

After the generation of the GVI and CI, the team is ready to begin making changes to the product architecture to develop a product platform more easily leveraged for future product generations.

Before continuing, it is important to remember the significance of each of the indices. The GVI is an indicator of the expected amount of redesign required for a component to meet the future market requirements. The Coupling Index – Receiving (CI-R) is an indicator of how likely a component will change when other components are redesigned.

The Coupling Index – Supplying (CI-S) is an indicator of how likely changing a component will require redesign of other components.

In Martin (1999), a detailed heuristic is shown to help the team decide on which components to standardize or modularize in order to create a robust product platform architecture. For this paper, a much-condensed description is given in the next three paragraphs.

In general, the team would like to standardize all the components. This translates into a product that can meet all the market requirements without having to be redesigned. Since this is generally not possible from a technical standpoint, or because the unit cost becomes prohibitively expensive, some method to decide which components to standardize is needed.

For standardization, those components that have high design costs and high GVI's should be focused on first. Another consideration is to standardize high CI-S components since they have a high potential for causing changes in other components. Standardization involves reducing the GVI and CI-R to zero. This means that no external or internal couplings will require the component to change. Three approaches to reducing these indices are discussed in Step 4.

Components that can not be standardized and will need to change should be modularized. This means that when the components do change, they will not require a change in any of the other components. This modularization refers to geometric changes as well as changes to the signal, material, and energy flows of the component. Modularization of the component requires reducing the CI-S to zero. The approaches used to reduce the GVI & CI-R are also used to reduce the CI-S.

DFV Step 4: Develop product platform architecture

Up to this point, the DFV method has covered descriptive measures of the design. This step applies a prescriptive approach to improve the architecture of the product. It will help the team make decisions on a) how to rearrange the mapping between the physical components and functions, and b) how to define interfaces. These are points 2 and 3 of Ulrich's definition of architecture. We do not address point 1 since we assume a decision has been made on the basic functions of the design.

How to reduce GVI

To reduce the GVI for the product, the method for determining the index needs to be considered. The GVI is calculated based on the engineers estimating the redesign costs required to meet the future customer needs - *for the architecture currently being considered*. The ratings for the GVI were previously displayed in Table 3.

The team should inspect this matrix to determine how the GVI's might be reduced. This is done by *explicitly* listing the specifications linking the engineering metrics (EM) to the components. These EM/Component specifications are what causes the component to be redesigned.

For instance, the PPM metric increases from 6 PPM to 12 PPM. The team estimates that a partial redesign (a rating of "6") of the PCA/Firmware is required to meet this future target value. This is estimated based on the team's expertise. The engineers know that their current architecture concept would require updated specifications for the processing algorithm, processor speed, and amount of RAM due to the PPM requirement moving from 6 PPM to 12 PPM.

These EM/Component specifications are substituted into the GVI matrix for each node. Example specifications are shown in Table 7.

Table 7: Explicit listing of GVI specifications

Engineering Metrics	**Components**				
	PCA/ Firmware	**Print Cartridge**	**Carriage Sub-System**	**Input/ Output Paper Tray**	**Feed Sub-System**
Pages per minute (PPM)	Processor speed Amount of RAM Software coding	Nozzle angle Firing rate Ink viscosity Ink drying time	Carriage velocity Sensor resolution		Roller vel. Roller dia.
Dots per inch (DPI)	Processor speed Amount of RAM # of outputs SW coding	Firing rate Drop size	Motor resolution Sensor resolution		Motor resolution
Decibels (dB)		Firing rate Weight	Bearing friction Motor RPM		Gear backlash Roller force
Footprint (sq in)	Board length Board depth		Carriage length	Length Width	
Reliability (MTBF)	Transformer temp. SW coding		Motor temp. Motor duty cycle	# of cycles	Motor temp. Motor duty cycle
Unit Cost ($)	Processor Amount of RAM SW coding	Ink well mtl Assembly Yields Tolerances	Motor PCA Bearing tolerance	Mtl cost	Motor Roller mtl.

There are two different major approaches to reducing or eliminating the GVI created by these specification flows.

1) Remove EM/Component specifications by:
   (a) Rearranging the mapping of functionality to components
   (b) "Freezing" the specification
2) Reduce sensitivity of the component to a change in the specification by:
   (a) Reducing the internal coupling within the component (i.e., within the CV)
   (b) Increasing the "headroom" of the specification

Approach 1 – Remove EM/Component specifications

1a) Rearrange the mapping of functionality to components

One approach to reducing the GVI is to change the architecture of the product to remove EM/Component specifications. Rearranging the mapping of the functionality to the components can do this. Consider the PPM/PCA-Firmware node, moving the processing of the print data to the computer's processor and RAM might reduce the

GVI. This would remove the specifications of "processor speed" and "amount of RAM". Such a change would also reduce the Cost and MTBF ratings by not requiring costlier and potentially less reliable processors.

Another rearrangement of the architecture might look at the “firing rate” required of the Print Cartridge nozzles. If the nozzles can not fire at a high enough rate to meet future DPI and PPM requirements, then the design of the Print Cartridge will have to be updated. One rearrangement is to use more nozzles rather than an increased firing rate to meet future needs for PPM and DPI. This could be accomplished by adding a larger print head to the Print Cartridge for each new generation requiring it. If the design is able to be modularized so that the update of the print head requires no other changes, then this would reduce the GVI.

1b) "Freeze" the specification

A pseudo-method for removing a specification is to standardize (or “freeze”) it. By freezing the specification, the team dictates that it will not be modified and thus has no possibility to cause other components to be altered. For the printer, the Print Cartridge ink drying time could be frozen at a certain value, thus “eliminating” it from the Cartridge / PCA and Cartridge / I-O Tray couplings.

Note that it will be difficult to standardize a specification that is tightly linked with customer needs. This is because it will constrain the team's ability to meet future needs. Also, there is always uncertainty in this method because while the team can state the specification will not change, there is always a possibility it will. Before freezing a specification, the team needs to fully understand the specification's relationship to the customer needs as well as how it is internally coupled within the component.

Approach 2 – Reduce sensitivity of the components to changes in the specifications

2a) Reduce internal coupling (within the component CV)

One reason for a component requiring lots of redesign is due to its own internal coupling. A small modification to a component can ripple throughout that component if the individual parts or features comprising it are highly coupled. For instance, if a changing customer requirement requires a change in the PCA, it might only require a change in a capacitor. However, if that capacitor is highly coupled with other capacitors, resistors, processors, etc. on the board, then the PCA has high internal coupling, and thus would require more redesign (a higher GVI) than if it were less coupled.

Another example is for the "firing rate" of the Print Cartridge described above. The high GVI rating resulted from the high internal coupling within the Print Cartridge. A change in the firing rate may require changes in the nozzle, the ink, the heat sinking, etc. A reduction in this coupling could reduce the GVI for that component.

The PCA is also an example of how redefining the internal coupling may reduce the requirement for redesign of that component. If the main processor used on the PCA is coupled to other electronic parts on the PCA such as diodes, capacitors, or other processors, then a change in the processor can ripple out to require changes in these other components, thus increasing the sensitivity of the components. This may require a total board redesign.

2b) Increase the "headroom" of the specification

Another method to reduce sensitivity is to increase the “headroom” of the EM/Component specifications. This implies designing the product so the component can absorb a larger change in the specification before requiring redesign. This is sometimes referred to as “overdesign”.

For the firing rate example, the team might develop a design that will enable higher firing rates (even though these rates are not needed for the current product). If this is accomplished then a change in this specification due to changing customer requirements will not require a redesign of the component. This will decrease the GVI.

Another example is with the MTBF of electrical parts. As the operating temperature of these parts approaches their maximum rated temperature, their reliability (MTBF) can decrease. One way to increase the MTBF is to lower the operating temperature of the component by increasing the size of the part (to better dissipate the heat). In this case the team needs to allow headroom in the geometry of the PCA. That is, space needs to be left to accommodate these larger components. If there is no space available, then a re-layout of the board is required.

A possible disadvantage of increasing specification headroom is that material costs may be increased. Also, it may not be possible to increase the headroom due to technological challenges for the current product.

How to reduce the Coupling Indices

Reducing coupling follows a similar approach to reducing the GVI. The team focuses on removing Component/Component specifications, or reducing their sensitivity. The CI matrix is redisplayed in Figure 19.

	Components supplying specifications										
Components receiving specifications	PCA / Firmware	Sensitivity	Print Cartridge	Sensitivity	Carriage Sub-Assbly	Sensitivity	I/O Paper Tray	Sensitivity	Feed Sub-Assembly	Sensitivity	**CI-R**
PCA / Firmware			Resistance # of nozzles pitch # of inks Firing rate Ink viscosity Drying time	9 3 3 3 3 3 3	speed Resistance Torque	3 6 1			Motor speed Resistance Torque	3 6 1	**47**
Print Cartridge	Voltage Firmware	6 1			dimension dimension dimension	6 6 6			resolution	9	**34**
Carriage Sub-Assembly	Voltage Firmware	6 1	Weight dimension dimension dimension	3 6 6 6							**28**
Input / Output Paper Tray			Drying time	3					Roller diameter X dimension Z dimension	9 3 3	**18**
Feed Sub-Assembly	Voltage Firmware	6 1	pitch	6			Z plane dimension dimension	6 3 3			**25**
CI - S	**21**		**57**		**28**		**12**		**34**		**152**

Figure 19: Complete CI matrix

Approach 1 – Remove Component /Component specifications

1a) Rearrange the mapping of functionality to components

Removing Component / Component specifications is one method of reducing the CI. If all specifications were removed, then there would be no coupling between the components. This is not possible since if a component is not coupled in some way to the rest of the components, then it is not contributing to the product.

However, selective specifications can be removed to help reduce coupling and thus slow down the propagation of changes throughout the product. For the printer, separating the ink cartridges from the print head and moving it to the chassis (this is referred to as an off-axis cartridge) removes the dimensional specifications between the two components. However, such a new architecture arrangement will also create new coupling specifications and these have to be considered.

Approach 2 – Reduce sensitivity of the components to changes in the specifications

2a) Reduce internal coupling (within the component CV)

Modifications in the internal coupling of a component can also help reduce the sensitivity of that component to shifting specifications, just as with the GVI. For instance, the internal coupling of the PCA could be modified so that a change in output voltage would not require any redesign of other components. This would reduce the effect of the Print Cartridge's resistance specification on the PCA since a change in the resistance could be accommodated more easily. This lowers the coupling index.

2b) Increase the "headroom" of the specification

For the printer, moving to the off-axis ink cartridge effectively reduces the sensitivity of the weight specification between the Print Cartridge and Carriage Sub-System. Since the ink was a large percentage of the weight of the Cartridge, now even large percentage increases in the rest of the Cartridge will not affect the Carriage. However, the new architecture arrangement can create new coupling specifications and these must be considered.

Overdesigning the receiving component to accept large specification increases can also increase headroom. The Carriage Sub-System can be designed to handle large weight increases in the Print Cartridge. The I/O Tray can be designed such that increases in drying time of the ink can be accommodated. The Feed Sub-System can be designed with an encoder resolution that will accept the nozzle pitch requirements for future products.

This increase in headroom will reduce both the CI-S and CI-R of the components. One thing to consider in this approach is the uncertainty surrounding the expected values of these specifications. The team could estimate incorrectly and then a specification may change enough to require a change in the receiving component. This could create large redesign efforts unless the receiving components internal coupling has been lowered.

5. CONCLUSIONS

The concepts and details of the DFV method have been used to aid in the design of desktop robots and network enclosure boxes with companies such as Sony and Nortel Network. An application of the method to the design of a thermoelectric water cooler demonstrates in detail how redesign efforts can be reduced using the method (Martin 1999). In addition, a major Japanese design and manufacturing firm and a U.S. electronics company are seeking to integrate the DFV concepts and methods into their product development process. The method is also an integral part of Stanford's DFM curriculum.

The need to develop products faster continues to be a major goal for many companies, and architecting is becoming crucial in helping companies accomplish this. This paper gives a general overview of a method designed to help teams reduce development time for evolutionary designs. The method determines those components that are most likely to change based on expected future market needs. It also determines parts that are coupled tightly with other parts. These insights taken together can lead to designs that minimize future efforts.

ACKNOWLEDGMENTS

This work was sponsored by the Stanford Integrated Manufacturing Association and the Department of Energy. The authors would like to thank Warren Hausman, Mark Steiner, Ab Stevels, and Steve Kmenta for their valuable comments on our research.

REFERENCES

Adler, P. S. (1995). “From project to process management”. *Management Science* **41** (3): 458-484.

Adler, P. S. (1996). “Getting the most out of your product development process”. *Harvard Business Review* **74** (2): 134-152.

Erens, F. J. (1996). *The synthesis of variety: Developing product families* Thesis. Eindhoven University of Technology**.** Eindhoven, Netherlands. 309 pages.

Fujita, K., S. Akagi, et al. (1998). "Simultaneous optimization of product family sharing system structure and configuration". *1998 ASME Design Engineering Technical Conferences*, Atlanta, GA. ASME New York NY USA.

Galsworth, G. D. (1994). *Smart, simple design : using variety effectiveness to reduce total cost and maximize customer selection*. Essex Junction, Vt., Omneo.

Gonzalez-Zugasti, J., K. Otto, et al. (1998). "A method for architecting product platforms with an application to the design of interplanetary spacecraft". *1998 ASME Design Engineering Technical Conference*, Atlanta, GA, USA. ASME New York NY USA.

Hauser, J. R. and D. Clausing (1988). “The House of Quality”. *Harvard Business Review* **66** (3): 63-73.

Krantz, D. H. and P. Suppes (1971). *Foundations of measurement*. New York, NY, Academic Press.

Martin, M. and K. Ishii (1996). "Design for variety: A methodology for understanding the costs of product proliferation". *1996 ASME Design Engineering Technical Conference*, Irvine, CA, USA. ASME New York NY USA.

Martin, M. and K. Ishii (1997). "Design for variety: Development of complexity indices and design charts". *1997 ASME Design Engineering Technical Conference*, Sacramento, CA, USA. ASME New York NY USA.

Martin, M. V. (1999). *Design for variety: A methodology for developing product platform architectures*. Mechanical Engineering Thesis. Stanford University**.** Stanford, CA. 172 pages.

Pine, B. J. (1993). *Mass customization : the new frontier in business competition*. Boston, Mass., Harvard Business School Press.

Robertson, D. and K. Ulrich (1998). “Planning for product platforms”. *Sloan Management Review* : 19-31.

Sanderson, S. and M. Uzumeri (1995). “Managing product families”. *Research Policy* **24** (5): 761-782.

Sanderson, S. W. (1991). “Cost Models for Evaluating Virtual Design Strategies in Multicycle Product Families”. *Journal of Engineering & Technology Management* **8** (3,4): 339-358.

Tseng, M. M. and J. Jiao (1998). "Design for mass customization by developing product family architecture". *1998 ASME Design Engineering Technical Conference*, Atlanta, GA, USA. ASME New York NY USA.

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

Wheelwright, S. C. and K. B. Clark (1992). “Creating Project Plans to Focus Product Development”. *Harvard Business Review* **70** (2): 70-82.

Wheelwright, S. C. and W. E. Sasser, Jr. (1989). “The New Product Development Map”. *Harvard Business Review* **67** (3): 112-125.

Yu, J. S., J. P. Gonzalez-Zugasti, et al. (1998). "Product architecture definition based upon customer demands". *1998 ASME Design Engineering Technical Conferences*, Atlanta, GA, USA. ASME New York NY USA.