Proceedings of IMECE2005
2005 ASME International Mechanical Engineering Congress and Exposition
November 5-11, 2005, Orlando, Florida USA

IMECE2005-82886

PRODUCT PLATFORM DEVELOPMENT: CONSIDERING PRODUCT MATURITY AND MORPHOLOGY

Tae G. Yang
Kurt A. Beiter
Kosuke Ishii
Department of Mechanical Engineering
Stanford University
Stanford, CA 94305-4021

ABSTRACT

In this paper we continue our development of a structured methodology for product platform development suited to the early design stages. In addition to the previous focus on differences between new and mature products, we now consider how platform development is influenced by the morphology of a product. Purely physical products (e.g., a vacuum cleaner) often pose fewer challenges when undergoing platform design. However, products such as software, service offerings, or process-oriented solutions require adapting previous platform methods in order to accommodate the "amorphous" nature of the system. This paper includes a review of the recent literature in this area, and presents the authors latest platform methodology from the perspective of product morphology and maturity. Illustrative examples are used throughout.

Keywords: platform-based design, modularity, design for variety

1. INTRODUCTION: BACKGROUND & MOTIVATION

Creating effective product platform designs continues to be a challenge in product development. While conventional hardware product platforms are becoming much more commonplace, software, service, and hybrid type systems continue to pose challenges to systems engineers in terms of developing an efficient platform. This is especially true for systems that are new or revolutionary in form and/or function.

Stanford's Manufacturing Modeling Laboratory (MML) continues to collaborate with a variety of commercial product industries on the topic of modular design for efficient platforms. Our collaboration with our industry partners has reinforced the notion that platform design in the context of non-traditional, or amorphous, products is the key to establishing lower overall development costs in an ever increasing competitive marketplace.

Academia has continued to produce a substantial amount of work on product family (PF), platform design, and, in a larger scope, modularity. Methodologies ranging from a generic framework to specific goal-driven optimization techniques offer means to achieve efficiency in platform design. However these studies generally consider physical product examples often involving geometric design parameters or relatively easily quantifiable metrics. Our own efforts seek to augment these approaches with a general framework that will accommodate these non-traditional amorphous products (Figure 1).

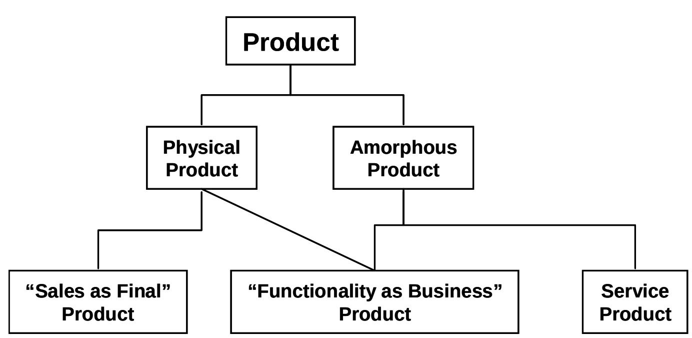

Figure 1: Different definitions of product based on specific business models

In this paper, the authors include an updated review of the recent work from academia and its implication towards practical application to industry towards platform design, and a proposed second generation of a generic framework that outlines a platform design methodology.

2. PREVIOUS WORK

2.1 Modularity

The concept of modularity is closely related to the platform architecture design. However, because of its wide scope, it has been discussed in numerous literatures at different levels of applicability. Table 1 highlights the 3 major modularity methodologies applicable at product design level.

Table 1: Three Major Modularization Methodologies

Methodology	Proposed by	Approach & Merits
Function Structure Heuristic Method	Stone et al.	•Function & heuristics based partition •Designers to make partition decisions
Design Structure Matrix (DSM)	Ulrich and Eppinger	•Subsystems and interface relationship in a matrix form •Good visualization of modules •Applicable to any domain •DSM output suggests partition
Modular Function Deployment (MFD)	Erixon	•Guided module driver identification •Use of QFD mechanism •Designer provided strategic insights to make partition decisions

Whether it is function based, interface requirement driven, or module driver dictated, all three methodologies provide valuable insights to the designer about how the partition should be made to better manage the complexity of a system. When a system is modularized efficiently, identification and improvement of system platform elements would be easily achieved. However, modularity alone does not contribute to identification and improvement of core platform elements. Thus, in this section, we focus our benchmarking of the notable work done in academia on platform methodologies.

2.2 Platform

Simpson [2003] provides an extensive review of the research activity that facilitates product platform design and optimization. He recognizes that product platform design is key to a successful product family, which many companies are utilizing to increase variety, shorten lead-times and reduce cost. Also, his definition of platform extends beyond that of module-based product families. He introduces scale-based product families with industry examples in which dimensional “stretching” allows for product variety. Depending upon product’s nature, an appropriate definition can be applied to a core set of product specific base solution elements.

Regardless of whether the platform is module-based or scale-based, the platform leveraging strategies can be mapped on the market segmentation grid that Meyer [1997a, 1997b] introduced (Figure 2). This grid lays out potential product positioning schemes within potential market domains, and helps companies to make strategic level decisions on how their platform should be designed – i.e., whether or not to pursue a module-based or scale-based platform strategy. Ishii and Martin [2000] also discuss such strategy in terms of “generational” and “spatial” varieties as the first step towards design for variety (Figure 3). This is a very critical step, as understanding the market and positioning your product appropriately would dictate the success of the designed platform. While these charts are extremely insightful for business level platform decisions, these two strategy charts alone do not provide specific instructions on how to form such a strategy, but rather, serves as guidance for platform design activities given a clear understanding of the market & strategy.

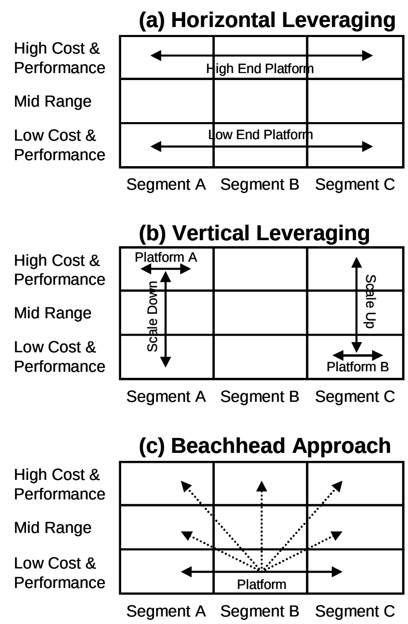

Figure 2: Platform Leveraging Strategies by Meyer

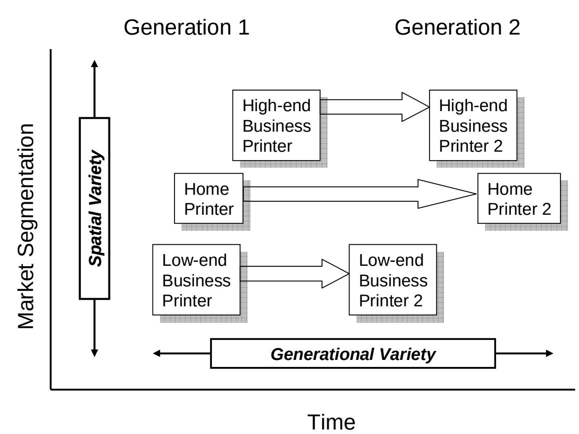

Figure 3: Spatial and Generational Variety by Ishii and Martin

In terms of platform design methods, numerous notable research activities have been reported as summarized in Table 2. Meyer & Lehnerd [1997c] introduce Composite Design as the process for defining platform strategy in support of their Market Segmentation Grid (Figure 2) as a step by step guideline for management. Their discussions are mainly on market, competition and cost-related metrics, and assume that a product's function structure map is readily available for competitive benchmarking. This method, in conjunction with the Market Segmentation Grid, provides the strategic guidelines for selection of which subsystems should be chosen to be the platform base, as well as which subsystems should receive more development attention for detailed optimization.

Ishii and Martin [2000] proposed Design for Variety (DFV) as a methodology for developing product platform architectures using Generational Variety Index (GVI) and Coupling Index (CI). Their approach provides a systematic method to capture customer needs at the product specification level, and is particularly helpful in the case of a physical product with a pre-defined structure and a well understood market.

Simpson et al. introduced Product Platform Concept Exploration Method (PPCEM) [2000a] for designing common product platforms that can be scaled or stretched into a suitable family of products. Simpson et al. also introduced Variation Based Platform Design Method (VBPDM) [2000b], a two-stage methodology for platform selection which is aimed at satisfying a range of performance requirements using the smallest variation of the product designs in the family. These methodologies are efficient in optimizing the commonality goals, and platform specifications, when given a geometrically pre-defined physical product designs.

In recent years, Bryant et al. [2004] reports a function representation based modularity approach to correlate functional modules to physical modules. They introduce Elimination Preference Index (EPI) as a metric to identify those modules and components that require redesign consideration. Kurtadikar et al. [2004] introduced a customer needs based methodology suitable for conceptual design phase. Based on similar function based modularity approach by Bryant et al., they use frequency and weight of customer needs to determine the common platform versus differentiating modules. Van Wie et al. [2004] presents platform element examination methods, based on the notion that a platform tends to be more integrated and differentiating elements tend to be more modular.

Otto et al. also has published a significant amount of practical applications in function-based modularization approaches in product platform/architecture design [2000a, 2000b, 2001a, 2001b, 2003]. While their examples are based on a physical product (document system) with pre-defined functionalities and structures, this methodology demonstrates the effective usage of modularity (function heuristic approach) in platform architecture decisions. Recently, Otto et al. [2004] reported a Multi-criteria Framework for Screening Preliminary Product Platform Concepts, which is focused on the early platform architecture phase. In this methodology, six categories of metrics, namely, customer, variety, flexibility, complexity, organization and after-sale, are proposed, which are difficult to evaluate in practice.

2.2 Platform Research Summary

The previous section introduced several platform related research activities. Table 2 summarizes the platform design related research presented in the previous sections. Overall, the leveraging strategies utilized by Simpson et al., Allen et al., as well as Ishii & Martin (Figure 3) appears to be the very first and critical step towards product platform design, since it will determine the architecture of the platform as well as associated design strategies. Fadel and Maier [2001] also recognize that determining which type of product family is the first strategic decision of product family design.

In terms of the methodology, Meyer & Lehnerd, Ishii & Martin and Otto et al. take a relatively qualitative approach in their methodologies, while others propose very quantitative optimization-based approaches. When the methodology is based on qualitative procedures, there is much room for designers to customize the methodology to suit their product development needs; but designers must also understand the intangible issues involved before making any decisions, and, such flexibility may introduce variability in results depending upon who executes the methodology. Quantitative optimization-based approaches may provide more consistent and repeatable results once the problems are defined for analysis. However, we believe the major challenges are in problem definition stage, where some of the critical issues maybe overlooked or underestimated.

All methodologies reviewed in this paper successfully demonstrate the claimed benefits using appropriate examples. We note that all—and the others that were not reviewed in this paper—employ 1) physical products, 2) geometrically quantifiable designs, and 3) pre-defined physical or functional structure. We would like to extend the scope of our methodology towards amorphous products, which may or may not have quantifiable design parameters or physical/functional structures.

Table 2: Platform Methodologies Recently Proposed by Academia

Authors	Methodology	Approach	Optimization	Example	Risk	Comment
Meyer & Lehnerd	Composite Design	Market Grid, Competition Benchmarking based to Generic Methods	No	Physical Products from Variety of Industry	No	Business level guidelines Assumes system structure & competition information are available.
Ishii & Martin	DFV	Customer Requirements → GVI, CI → Product Specifications	No	Printer	No	Assumes valid customer requirements
Simpson et al.	PPCEM & VBPDM	Leveraging Strategy →Optimization → Product Specification	Yes	Universal Motor	No	Quantitative metrics & relationships needs to be defined
Otto et al.	Function Based Modularity	Function Structure → Modules → Product Portfolio	No	Document System, Drills, etc.	No	Assumes function structure available
Allen et al.	Multiple Platform Selection	Leveraging Strategy → Optimization → Iteration → Product Portfolio	Yes	Absorptio n Chillers	No	Quantitative metrics & relationships needs to be defined
Allada & Jiang	Robust Platform	Changing Customer Requirements → Optimization/Simulation → Product Specification	Yes	Vacuum Cleaners	Yes	Assumes function-module mapping available
Bryant et al.	Modular Design Approach to Support Sustainable Design	Functional Represntation→3 heuristics→potential modules→Functional module to physical module mapping→Elimination Preference Index (EPI)	No	Hand-Vacuum	No	Redesign of high-EPI modules & components
Kurtadikar et al.	Customer-needs Motivated Conceptual Design	Function based approach→Frequency & weight of Customer needs → Platform vs. differentiating modules	No	US Army Battlefield Hazard Marker	No	Suitable for conceptual design stage
Van Wei et al.	Platform and Differentiation Elements in Product Design	Integrated platform vs. modular differentiating modules	Yes	Single use cameras	No	

3. OUR APPROACH

Our approach includes the 5 major steps shown in Figure 4. These steps emerged through our work on several case studies from industry that included both physical products (e.g. notebook computers) as well as amorphous products (e.g. a new medical service as a business). In our approach, we categorize platform development projects into two categories based on the product maturity level: *Platform Innovation* and *Platform Refinement*. Regardless of whether the product is new or mature, the 5 major platform development steps will still apply; however, there are differences in tool and method application based on product maturity level, and these differences will be discussed in the following sections.

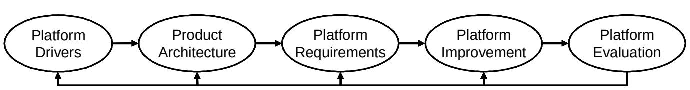

Figure 4: Important steps in platform design

Through numerous case studies with companies such as Toshiba, Ebara, ABB, and Nissan, Stanford's MML has observed that different businesses require different types of platforms, and such differences could be better identified through thorough examination of a company's business model. Our 5 step methodology provides a general approach, and we specify when and how to use each step for different products and their platforms. In order to provide clearer guidance, the following sections detail our definitions of a product and platform. We also present a Product Morph-Maturity Map, which can be useful in understanding a company's platform development strategy.

3.1. DEFINITIONS: Product & Platform

The purpose of this section is in providing a common platform development language across industries, and this language will be used throughout our methodology. We feel that this is necessary since different industries' understanding of products and platforms may be quite different.

A *product* is a collection of features that:

- entail complete set of functionalities perceived as valuable to customers in a given market under specific time frame (product life)
- can be either physical or amorphous, or a blend of either
- are the means upon which companies realize revenue in exchange

While there are many other definitions reported in academia, our discussions on platform will be limited to the following characteristics: A *platform:*

- is a base architecture upon which market – or time-specific – functionalities are built
- may not entail complete functionalities
- may be a combination of fundamental key solution elements
- is driven by business strategy

3.2. Product Morphology-Maturity Map

Product Morphology-Maturity Map (Figure 5) consists of two axes: morphology level and maturity level. These axes are of a qualitative & relative scale. Each quadrant represents a product type with corresponding morphology and maturity level:

- Type-I is a New-Physical (e.g. body warming device of Company "D")
- Type-II is a New-Amorphous (e.g. medical imaging service of Company "T")
- Type-III is a Mature-Physical (e.g. notebook PC)
- Type-IV is a Mature-Amorphous product (e.g. software) groups.

These product groups are just a guideline, and there could be products that belong to multiple groups. For example, most aircraft engine manufacturers are combining the maintenance service (amorphous) with their engines (physical).

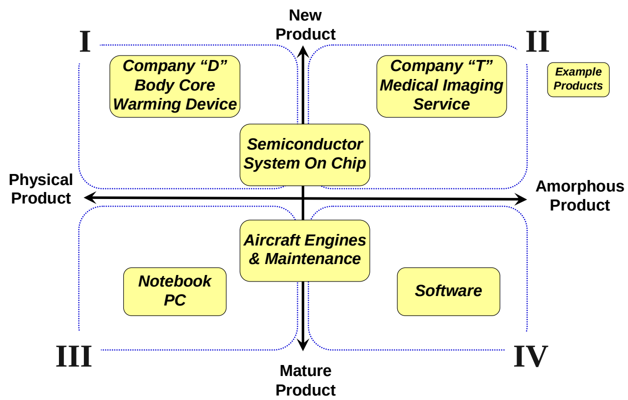

Figure 5: Product Morph-Maturity Map

A product's morphological characteristics impact the analysis techniques used throughout our platform development steps. Traditionally, the dfM tools we recommend in our steps are well suited for either typical physical products (Type-III), or mature amorphous products with clear modules (Type IV). However, when faced with unclear (i.e., as-yet-to-be determined form) and intangible amorphous products, describing and analyzing such a system becomes challenging. A functional description of an amorphous system can be dealt with by conventional dfM tools effectively. Yet, the majority of the MML's industry partners have expressed the need for appropriate provisions in current dfM tools, especially in tools that require system component/module descriptions. For example, the Function-Structure map and QFD-II require system descriptions other than functional hierarchy, and engineers are often challenged with describing an amorphous product's "physical" structure. For such cases, we have updated our platform development methodology and where necessary, modified our dfM techniques.

Based on the products' maturity level, we define two types of platform projects—*Platform Innovation* and *Platform Refinement:*

- Platform Innovation projects are for the products that fall into regions I & II of the map. Common challenges are market identification, voice of customer collection, initial product launch strategy, and formulation of financial analysis.
- Platform Refinement projects are for products in regions III & IV. Common challenges are inefficiency identification, strategic re-alignment in regards to core-competency, and justification of new investment.

We categorized platform projects into these two main types since the maturity level of a product has a direct impact on the execution sequence of our methodology.

3.3. Step 0: Understand your Product

Project development constraints such as development cost, time-to-market, and target performance levels all have an impact on platform development. This impact depends primarily on the relative priority of these constraints. The importance of establishing such a project development priority amongst features, cost and time was recognized by [Wilson, 1993], and we adopt the *Project Priority Matrix* methodology explained in this work to help plan a path through our platform development methodology.

The Product Priority Matrix [Wilson, 1993] is a 3-by-3 matrix that relates product performance (features), development cost (cost), and time-to-market (time), to project priorities described as *constrain*, *optimize*, and *accept*. The highest priority is considered to be constrained; for example, if there were a hard limit on time-to-market, then development *time* would be considered constrained. The next highest priority is considered as *optimize*, meaning, given the project constraint, this factor is optimized at the expense of the remaining factor, which is *accepted*. Such a *constrain-optimize-accept* approach to prioritizing project features, cost and time has been shown by [Wilson, 1993] to be a key indicator of project success or failure..

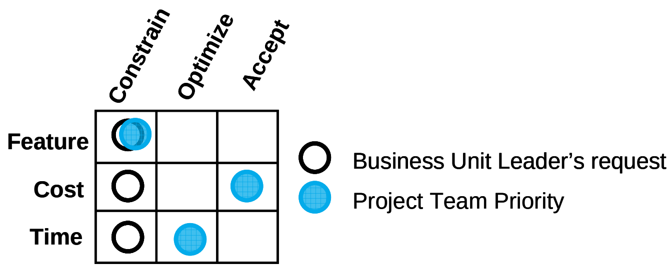

Figure 6: Product Priority Matrix Example

Figure 6 shows an example of a Product Priority Matrix, of a typical business unit leader's request vs. a correctly executed matrix by a project team. A project manager may typically desire to meet *fixed* project cost, time, and feature targets. In reality, such a scenario can often lead to project delay, cost over-run, or outright failure. A simple rule for this matrix exercise is that only one mark per column and per row is permitted in order to force project teams to identify their products' priorities. The example shown in Figure 6 is typical for a product in an emerging high tech market; in this case, the features of the product have the highest priority, the development time is prioritized second, and the development cost is considered the least important aspect of the product.

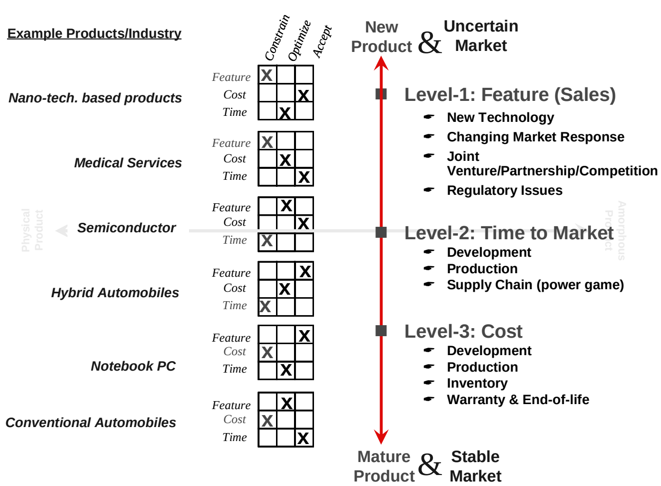

Figure 7: Platform Priority Matrix Guideline

The *Platform* Priority Matrix employs the same prioritization mechanism, although it differs from the original *Product* Priority Matrix in that we apply it at a business level based on a product's *maturity*. There are three levels (Features, Cost, and Time) with two possible priority configurations each. The first level is on the *new product* end of the product maturity axis, the second level is in the middle of the axis, and the third level is on *mature product* end of the axis. Figure 7 summarizes these three levels and highlights a few examples.

1) Platform Priority Matrix Level 1: New & Uncertain Market

In Level 1, a business is dealing with a new product with uncertain market responses. In this case, the Platform Priority Matrix will typically have *feature* as the constrained priority. Typical objectives in a *feature-constrained, new-product* project include transforming new technology into customer perceived value, and packaging the features for maximum market capture with minimum technological, competitive, and regulatory risk. Then, cost and time are adjusted according to the specific business situation.

2) Platform Priority Matrix Level 2: Semi-mature product & market

In Level 2, a business is dealing with an established product market with a good market response for a given product, and a few key competitors that are pushing for faster times-to-market in order to expand market share. In such a *time-constrained* project, fast development, production and supply chain management dominate the priority list. Features and cost are adjusted according to the specific business situation.

3) Platform Priority Matrix Level 3: Mature & Stable product & market

In Level 3, the product and market is mature and stable, and the most of the business practice is focused on cost reduction efforts. Reducing cost of development, production, inventory management, and warranty are the highest priorities for businesses at this level, and a platform that supports these cost reduction efforts is desired. Features and time are adjusted according to the specific business situation.

4. METHODOLOGY

4.1. Step 1: Clarify platform drivers

As the design begins, it is necessary to clarify the platform drivers for the product development project. The platform drivers address the system objectives at a high level and include such goals as lower overall development cost, reduction of costly product variety, faster times-to-market for new products in the family, etc. Figure 8 shows the three related aspects of this step: design drivers, dimensions of variety, and market segments.

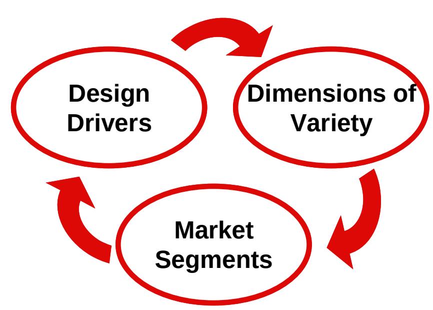

Figure 8: Platform drivers are related to the market, the product, and the organization

Typical design drivers include such items as:

- Cost
  - Direct/indirect, time/labor, material, capital
- Complexity
  - Number of subsystems, parts, processes
- Commonality
  - Similar parts, processes, materials, interfaces
- Features, Reliability and Quality
  - Performance, robustness, service life
- Competency
  - Core competency vs. outsourcing components and technology
  - "Make vs. Buy"
- Flexibility and Variability
  - Architectural "headroom"

The design drivers depend upon the market requirements, and in conjunction with the market needs suggest appropriate dimensions of variety. Conducting the Platform Priority Matrix exercise in Step 0 should confirm the drivers in conjuction with the results gained via the following *dfM* tools:

Customer Value Chain Analysis (CVCA): CVCA is a block diagram that includes various project stakeholders and flows of money, information, complaints, product, etc. Donaldson et al. [2004] explains CVCA as "an original methodological tool that enables design teams in the product definition phase to comprehensively identify pertinent stakeholders, their relationships with each other, and their role in the product's life cycle." By applying CVCA in this step, platform designers can identify the right sources of platform drivers, and can generate the appropriate set of variety dimensions. It also confirms the product's business model, and clarifies the value proposition to be embedded in the product.

Voice of Customer (VOC) Analysis: Once CVCA identifies the major stake holders, VOC collection/analysis identifies the key requirements of each stakeholder. Depending on the CVCA exercise, *customers* may include end users of the product, project team's managers (Voice of 'Boss'), and others such as regulatory agencies. During VOC analysis, an emphasis should be on variety requirements, which ultimately will aid in identifying common versus differentiating modules. Techniques such as market surveys, conjoint analysis, and affinity diagrams are useful. As a result of VOC analysis, a design team should have a prioritized list of variety requirements.

Quality Function Deployment (QFD) Phase-1: QFD-1 maps a prioritized list of Customer Requirements (CR) generated from VOC analysis, to an appropriate set of Engineering Metrics (EM). It produces a list of EMs with relative importance. For platform development, since the CRs represent variety requirements, QFD-1 can be implemented to obtain a prioritized list of variety dimensions (i.e. EMs).

The results from Step 1 of the platform development methodology are;

- Strategic Alignment: Platform Priority Matrix
- Major Stakeholders: CVCA
- Prioritized Variety Requirements: VOC Analysis
- Prioritized Variety Dimensions: QFD-I

4.2. Step 2: Identify the product architecture

This step of the process identifies the structural and/or functional elements that comprise the system, as well as insights to those components that will deliver the required variety and those components that will provide the basis for the platform architecture. Also, a company's core competency modules and relevant technology and/or processes should clearly be identified as major candidates for a platform.

In case of *Platform Innovation* projects (i.e. Type-1 and/or Type-2 projects), a preliminary concept generation step is required. Some projects may not even have a description of a complete system. By gathering the functional requirements from the VOC analysis in Step 1, Morphological Analysis [Pahl, 1996] should be carried out to roughly describe the intended new product's functionalities. For *amorphous* products (Type-2), solution elements can be substituted for physical modules. Then, Pugh Concept selection [Pugh, 1996] can be used to select a preliminary system concept by using the Variety Dimensions as a selection criteria. This preliminary description of a product's architecture will serve as a basis for the remaining steps, and puts *Platform Innovation* projects in the same position as the *Platform Refinement* projects for the further analysis using the following *dfM* tools.

Function-Structure (FS) Map: The function-structure map is a hierarchical functional diagram linked to a hierarchical structural diagram [Ishii et al., 2004]. By considering which functions are supported by which components, a designer can gain a better understanding of the system, identify key components that provide variety, and also extract a list of sub-functions that are critical to platform design.

Quality Function Deployment (QFD) Phase-2: QFD-2 maps the prioritized EMs to system components and modules (or, solution elements). With the structure information of an intended product from a function-structure map, a QFD-2 can be implemented to identify the relative importance of modules/solution elements as reflected by the VOC analysis from Step 1. The results from this tool will confirm the qualitative results from FS map, and provide quantitative data of product modules for further analysis.

Cost-Worth Analysis (CWA): CWA is a graphical tool, which plots cost-worth ratios (relative cost divided by relative 'worth' as computed in QFD-2) of a product's solution elements. If a cost-worth ratio were close to "1," this would imply that a module costs approximately the same amount of the value it provides to the product. If it were much higher than "1," it would imply that a module costs too much for its value, and vice versa. By plotting each module on a relative cost vs. relative worth graph, one can easily identify cost reduction and value enhancement candidates.

In the case of *platform innovation* projects, cost data may not be available. When a reasonable cost estimate is not feasible, other metrics can be substituted for the relative cost axis. Estimating the relative difficulty, or required resources such as required engineering hours and time to complete development can provide informative and essential insights to product's cost-worth structure for further analysis.

The results from Step 2 of the platform development methodology are;

- Creation of preliminary product structure (Morphological Analysis & Pugh Concept Selection for *Platform Innovation* projects)
- Hierarchical layout of product's functions & structure & the list of variety enabling modules (FS Map)
- Relative importance (worth) ratings of product's modules (QFD-2)
- Insights to cost-worth ratings of each module & company's core competency. (CWA)

4.3. Step 3: Identify module requirements vs. complexity

The third step in platform development establishes the relationships between platform requirements and complexity for the modules comprising the system. From step two, we have identified those modules which will be required to deliver variety. Ideally, modules that provide variety—i.e., modules that must be interchanged in order to create a specific instance of the platform—should have low complexity, which can be measured in terms of integration with other modules in the system, design lead-time, or overall design cost (see Ishii et al. [Ishii, 2004]; other measures are also possible).

Quality Scorecarding: Quality Scorecarding is a qualitative process of identifying project objectives (biggest Y), supporting metrics (big Ys), project alternatives (vital Xs), and uncertainties (noise, Vs). For example, most product development projects would have Net Present Value (NPV) or Return on Investment (ROI) as the biggest Y, metrics relating to revenue and cost as big Ys, various design choices as vital Xs, and project inherent uncertainties such as market response, manufacturing variation, etc., as Vs. By examining the project's Quality Scorecarding thoroughly, complexity factors and requirement factors may be identified & confirmed, especially in the big Ys. Examples of complexity factors for a typical product development project include (but are not limited to) development, supply chain, manufacturing, assembly, marketing, and service related cost, lead-times and steps. Typical variety requirements would be number of models and services required by the market and enabled by technology. Both the complexity and requirement factors are project dependent, usually reflecting the platform driver identified in Step 1 of this methodology.

Design for Variety (DFV) Charts: These charts plot product module requirements against product module complexity. Figure 9 shows a generic DFV chart, and indicates the region in which product modules should be improved (the high-variety-requirement, high-complexity region of the graph; or, upper right-hand portion of the graph). The general governing principle is that modules with both high variety requirements and high complexity are costly to design, manufacture and support, and lead to overall higher costs of providing product variety. Therefore, modules in this region of the graph should be redesigned to ensure lower complexity or redesigned to more simply provide the needed variety. The diagonal line in Figure 9 serves as a generic guideline to divide the High Priority Region from the Low Priority (or Desired) Region.

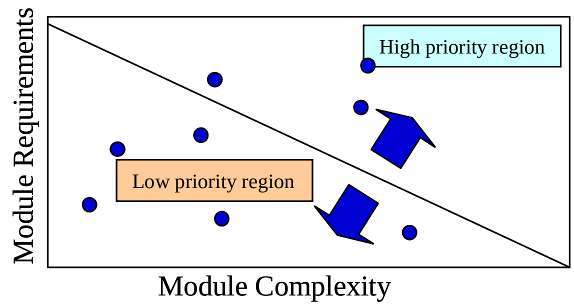

Figure 9. General Format for a DFV Chart

The y-axis (module requirements) is some measure of required modularity for either providing variety or serviceability. Typical measures include Variety Voice of the Customer (VVOC), anticipated Generational Variety (GVI) [Ishii & Martin, 2000], and Service Frequency. The x-axis is some measure of product complexity, typically captured in terms of development effort (time or cost), production lead time, number of service steps, etc. Choice of requirements and complexity measures should capture the most important metrics identified in the Quality Scorecarding exercise (see above).

As an example, consider the DFV chart for an inkjet printer (shown in Figure 10). The y-axis is required variety (VVOC), and the x-axis is manufacturing lead-time. In the high priority region, two modules are candidates for improvement. One is the drive mechanism with print head, and the other is power module.

The DFV chart identifies potential modules for improvement as well as modules suitable for platform elements. In the inkjet printer example, assuming that lead-time is the major complexity factor, the tray, electrical system, housing, power, print mechanism and base modules are good candidates for platform elements. The remaining modules can provide the required variety in performance. This information is used in Step 4 of the methodology for generating ideas to improve the platform design.

Sub- Assy Name	Lead Time*	Variety**
Base	0.9	0.1
Housing	0.4	0.2
I/O Tray	0.2	0.6
Electronics	0.3	0.3
Power Supply	0.8	0.8
Mechanism w/Printhead	0.6	0.72
Ink	0.1	0.89

*Lead Time in relative scale

** Variety index
"0" = common across all models
"1" = different from model to model

Figure 10a. Variety vs. Lead-Time data for an inkjet printer

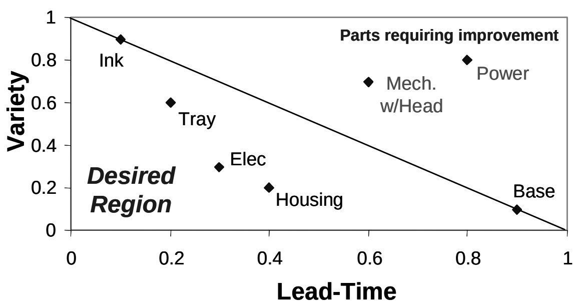

Figure 10b. Variety vs. Lead-Time DFV chart for an inkjet printer

In summary, the results from Step 3 of the platform development methodology are;

- list of complexity & requirement metrics (Quality Scorecarding)
- DFV chart with appropriate complexity and requirement metrics
- candidate modules for improvement

4.4. Step 4: Generate ideas for platform improvement

This step uses the results from step 3 to focus on opportunities for re-design. The main activities in this step are concept generation via Morphological Analysis, and concept selection via Pugh Concept Selection.

Morphological Analysis: Morphological Analysis, as described by Pahl and Beitz [1996], is a systematic study to analyze the possible shape and form for a set of Morph keys generated from essential functions of a given system. For the purpose of this methodology, the Morph Keys can be selected from the analysis results of the previous 3 steps. Important VOCs, engineering metrics from QFD-2, cost driver solution elements, and complexity/requirement axes (illustrated in the DFV chart) are all good sources of Morph Keys. For each key, a variety of possible solutions can be brainstormed, and when completed, a selected combination of the solution ideas forms a new system concept. We recommend that a few concepts with varying degrees of technical feasibility, risk, and cost be chosen for systematic selection process via Pugh Concept Selection technique.

Recall that the inkjet example (from step 3) identified the power module and drive mechanism with print head modules as targets for redesign. A potential redesign idea for the power module includes accommodating different voltage inputs across different markets, thus reducing the variety requirement. A potential redesign idea for the drive mechanism with print head includes dividing the drive mechanism and the print head into two modules; this accomplishes the following objectives: the variety requirements are supported in the much simpler print head, and the more complex drive mechanism is commonized across the product line (Figure 11). In this case, longer lead times for a common item are more easily managed since the part is common across the product line; and, the shorter lead-times for the print-head will be enable faster response to unforeseen changes in sales volume for a particular print-head variety.

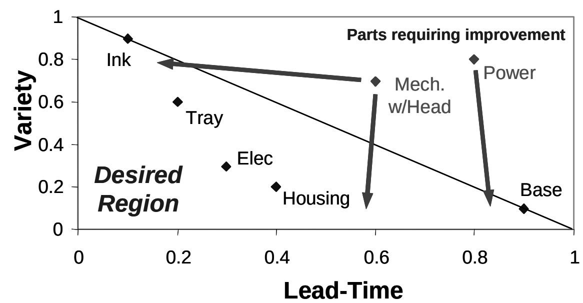

Figure 11. Two ways to move high priority modules to the desired region of a DFV Chart

Pugh Concept Selection: Pugh Concept selection [Pugh, 1996] is a systematic way of selecting the best concept among candidate concepts with a given set of selection criteria. The selection criteria can be drawn from the Quality Scorecarding exercise in Step 3; this is especially true of the Big Ys, which represent the performance of an intended system. Candidate concepts and the criteria are tabulated in matrix form, and each concept is evaluated with comparison indicators ('+,' '-,' or 'S' for 'Same') for each criteria against a selected datum concept. An iterative process of alternating the datum concept leads to identification of the best concept.

In summary, the results from Step 4 of the platform development methodology are;

- List of Morph Keys & candidate concepts (Morphological Analysis)
- List of concept selection criteria and the best concept for an intended platform (Pugh Concept Selection)

4.5. Step 5: Evaluate the net present value

The focus of step 5 is the evaluation of the platform strategy. An attractive overall platform metric is some measure of cost/benefit, since it can capture not just the expected return on the design of a single instance, but capture the integrated effect on the entire platform family. Figure 12 shows qualitatively the expected benefits of adopting a platform approach for a product family. While up-front investment may be larger due to increased base platform design effort, future derivative product releases should require lower investment.

Net Present Value Analysis (NPV): NPV analysis in nature is unique for different businesses, and may impose challenges for *Platform Innovation* projects because the immaturity of the product may not have established an accurate way of estimating the NPV. Also, the projects that identified very large uncertainties in their Quality Scorecarding (Vs) may be subject to large variation in their estimates. In order to address these risks associated with either lack of data, or known uncertain events, we recommend a Decision Analysis approach.

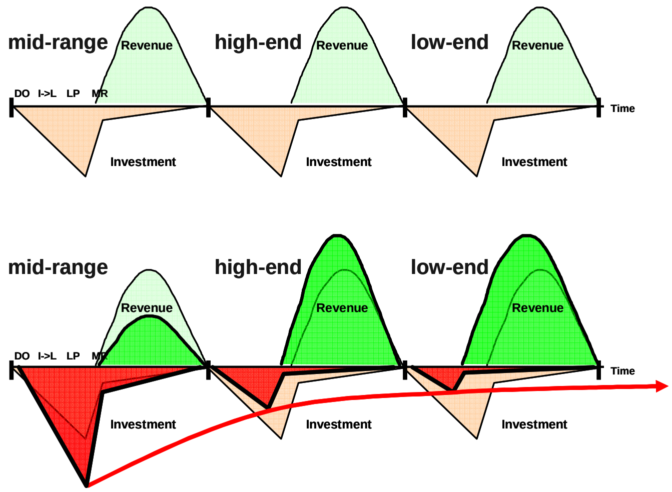

Figure 12: Generational investment/revenue graph for traditional design (top) and platform (bottom)

Decision Analysis (DA): Decision Analysis [Howard, 1993] is a normative approach to improve decision quality by addressing the uncertainties with systematic methods. Yang et al. [Yang, 2005] showcase a DA-based educational case study which simulates the dynamic nature of platform development process. Also introduced are a few key tools from DA, namely the Decision Influence Diagram and the Decision Tree. Refer to [Yang, 2005] for a detailed description of these tools. The key to a DA-based approach is distinguishing the *uncertainties* from the generic *risk* and *ambiguity*, and analyzing their impact on each decision alternative. For this platform methodology, a DA-based approach may serve as a means of reaching a 'go/no-go' decision regarding specific platform implementation.

4.6. Methodology Summary

There are three merits to our methodology: generic, guide, and complete. The methodology can be applied to a wide variety of products regardless of shape, maturity, and/or business model. The product morphology-maturity map provides a relative reference and suggests a strategic direction for an intended platform. The step-by-step nature of the methodology serves as guidance for project flow (Figure 13), with suggested dfM tools and varying execution techniques for different product types. Also, an estimation of NPV in the last step completes the platform development activity with a single bottom line metric that upper management can make decisions upon.

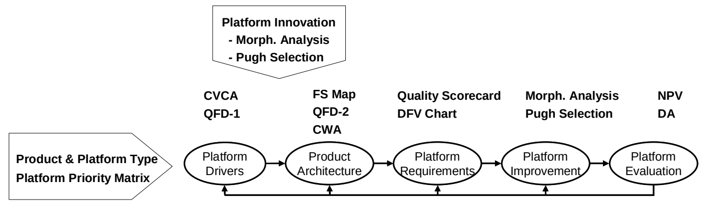

Figure 13: 5-Step Platform Development Methodology Summary

5. CONCLUSIONS & FUTURE WORK

In this paper, we presented the Morphology-Maturity map as guidance for companies to confirm their business model and to make strategically sound platform decisions. Linked with such strategic understanding of one's business and product, the 5 steps of the platform development methodology shows how to utilize existing dfM tools to address even the most unconventional product domains that recent platform studies from academia have not covered.

The methodology is generic enough so that different industries may adapt it with appropriate modifications. An integration of Decision Analytic fundamentals in Quality Scorecarding & NPV analysis completes the methodology so that high level performance metrics such as project risk & financial consequences can be quantified in relevant contexts. However, the methodology is most easily deployed within an industry if a variety of real-life examples drawn from the industry are piloted. In collaboration with companies like Toshiba, ABB, GE Transportation, Ebara, Nissan, and Hitachi, the Stanford MML is compiling case studies to demonstrate different implementation techniques and improve upon the methodology.

REFERENCES

Bryant, C.R., Sivaramakrishnan, K.L., Van Wie, M., Stone, R.B., and McAdams, D.A., "A Modular Design Approach to Support Sustainable Design," Proceedings of DETC 2004, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conferences, Salt Lake City, Utah, Sept. 28-Oct.2, 2004.

Donaldson, K.M., Ishii, K., and Sheppard, S.D., 2004, "Customer Value Chain Analysis," *Proceedings of ASME DETC: CIE*, Salt Lake City, Utah, USA.

Fadel, G. M. and Maier, J. R. A., "Strategic Decisions in the Early Stages of Product Family Design," Proceedings of DETC 2001, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Pittsburgh, PA, September 9-12, 2001.

Howard, R.,1983, "The Evolution of Decision Analysis"

Ishii, K. and S. Kmenta, 2004, "Value Engineering (Value Identification and Functional Analysis)" ME317 dfM:

Product Definition Course book, ed(s): K. Ishii. Stanford Bookstore, Stanford University

Ishii, K. and Martin, M.V., “Design for Variety: A Methodology for Developing Product Platform Architectures,” Proceedings of DETC 2000, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Baltimore Maryland, September 10-13, 2000.

Kurtadikar, R.M., Stone, R.B., Van Wie, M., and McAdams, D.A., “A Customer Needs Motivated Conceptual Design Methodology for Product Portfolios,” Proceedings of DETC 2004, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conferences, Salt Lake City, Utah, Sept. 28-Oct.2, 2004.

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

Meyer, M. H., 1997, “Realize Your Product Lines Through Continuous Platform Renewal,” Research Technology Management, 40(2), pp. 17-28.

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

Meyer, M. H., 1997, “Realize Your Product Lines Through Continuous Platform Renewal,” Research Technology Management, 40(2), pp. 17-28.

Otto, K.N. and Gonzalez-Zugasti, J.P., “Modular Platform-Based Product Family Design,” Proceedings of DETC 2000, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Baltimore Maryland, September 10-13, 2000.

Otto, K.N., Gonzalez-Zugasti, J.P., and Dahmus, J.B., “Modular Product Architecture,” Proceedings of DETC 2000, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Baltimore Maryland, September 10-13, 2000.

Otto, K.N. and Dahmus, J.B., “Incorporating Lifecycle Costs into Product Architecture Decisions,” Proceedings of DETC 2001, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Pittsburgh, PA, September 9-12, 2001.

Otto, K.N. and Sudjianto, A., “Modularization to Support Multiple Brand Platforms,” Proceedings of DETC 2001, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Pittsburgh, PA, September 9-12, 2001.

Otto, K.N. and Holtta, K.M.M., “Incorporating Design Complexity Measures in Architectural Assessment,” Proceedings of DETC 2003, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Chicago, Il, September 2-6, 2003.

Otto, K. and Holtta, K., “A Multi-Criteria Framework for Screening Preliminary Product Platform Concepts,” Proceedings of DETC 2004, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conferences, Salt Lake City, Utah, Sept. 28-Oct.2, 2004.

Pahl, G. and Beitz, W., Engineering Design: A Systematic Approach, Springer-Verlag, 1996.

Pugh, S. 1996, Concept Selection – A Method that Works, Chapter 14 from “Creating Innovative Products using Total Design,” Addison-Wesley, pp.167-176. ISBN 0-201-63485-6

Simpson, T.W., Martinez, M.P., and Messac, A., “Effective Product Family Design Using Physical Programming and The Product Platform Concept Exploration Method,” Proceedings of DETC 2000, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Baltimore Maryland, September 10-13, 2000.

Simpson, T.W., Chen, W., and Nayak, R.U., “A Variation-Based Methodology for Product Family Design,” Proceedings of DETC 2000, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conference, Baltimore Maryland, September 10-13, 2000.

Simpson, T.W., “Product Platform Design and Optimization: Status and Promise,” Proceedings of DETC03, ASME Design Engineering Technical Conferences and Computer and Information in Engineering Conference, Chicago, Il, September 2-6, 2003

Van Wie, M., Stone, R.B., Thevenot, H., and Simpson, T., “Examination of Platform and Differentiating Elements in Product Design,” Proceedings of DETC 2004, ASME Design Engineering and Technical Conference and Computers and Information in Engineering Conferences, Salt Lake City, Utah, Sept. 28-Oct.2, 2004.

Wilson, E., 1993, “Maximizing Designer’s Impact on Market Success through Product Definition,” *Design Management Journal*, Fall, pp.62-68.

Yang, T. G., Ishii, K., and Karandikar, H., ”A Decision Analysis Based Approach to Product Development Simulation Training”, to appear in the 2005 Proceedings of the ASME Design Technical Conferences.