Architecting and Innovating

Ronald B. Campbell, Jr.
Senior Lecturer
Center for Innovation in Product Development
Massachusetts Institute of Technology
30 Memorial Drive, Room E60-275d
Cambridge, MA 02139
Sandy_C@mit.edu

Abstract. Innovating is essential to sustained industrial growth and profitability. But experience amply demonstrates how difficult innovation is, especially for large companies. The synthesis of valued offerings by aligning customer needs with technology possibilities lies at the heart of innovation. System architects working at the strategic level are ideally positioned, as a consequence of their experience and training, to play a key and even a leadership role in enabling, energizing, and leading this synthesis. The scope of the architecting effort must include the process architecture of the entire value chain as well as the more conventional product architecture to address all potential wellsprings of innovation. This paper outlines an architecture-centric approach to innovation, based on the concept of the *system platform architecture.*

Introduction

"In architecting a new software program, all the serious mistakes are made the first day." (R. Spinrad quoted in (Rechtin, 1991.)) This apothegm captures the importance and leverage of the early work in the front end of software development. Extended to product development, it implies the importance of creating early on a robust and enduring base. Let us generalize Spinrad's apothegm to say, "In architecting a new platform, all the serious mistakes are made the first day." Our core proposition is that the *system platform architecture*, or top level design, is the initial expression of the synthesis of customer needs and new technology to enable and generate innovation, and that it is critical to get it roughly right, that is, to avoid serious mistakes, early on. Consequently the central role for the *strategic system architect* is to lead in conceiving, defining, and designing this platform.

For working purposes in this paper, we define system architecting as *the process of evolving the right system*, and system engineering as *the process of developing the system right*. System designing is the union of these two, and accordingly could be said to be *the process of getting the right system right*. This paper focuses on architecting during the front end to generate the best platform for market success.

The context of this paper is commercial product development, that is, the development of products for highly competitive markets with many buyers and a significant number of competitors. Product development for a single large acquisition authority (monopsonistic market) has too many rigidities and too much inflexibility to embrace the level of concurrency and flexibility mandated by the platform development process outlined here. See (Clausing et al 1999) for a fuller explication of this subject. Furthermore, we focus on innovations requiring

such large scale technology investment that a platform based approach is appropriate.

The next section addresses the subject of innovation. A working definition is provided, examples given, and key critical factors relevant to the system architecture are elucidated. In the following section, the structure and function of the *system platform architecture*, in particular how it aligns and engages customer needs with technology, are described. The system platform is the *union of the product platform, the process platform, and the economic platform*. A process for system platform development is outlined. The last major section looks at the role of the system architect, emphasizing personal capabilities, key activities, and positioning within the management system. To function effectively in the prescribed role, the architect must be an exemplar of integrated knowledge spanning business and technology, and furthermore must combine creativity with the drive to execute.

Innovating

Definition. We define innovation strictly as *the successful creation of new market value through the satisfaction of customer needs with new technology*. Bluntly, it isn't innovation until it succeeds in the marketplace. The central challenge is the alignment of new (or emerging) customer needs with new (or emerging) technology in the form of market offerings. Successful in this context implies financial returns that exceed investment guidelines based on revenue, share, and profit requirements. One belief in this arena is that the creation of new business value is essentially unfathomable, e.g. the "let a thousand flowers bloom" school, but our perspective is that there is much that can be done through *strategic architecting* to improve the probability of success.

There is a very large and rapidly growing literature on innovation, undoubtedly due to its importance in generating growth in an era of world-wide competition. We cite here a few references most germane to our thesis. Utterback's classic work on the dynamics of innovation (Utterback 1996) takes a long historical perspective and perhaps best covers the entire subject. Christensen's two recent works (Christensen 1997 and 2003) are exceptionally valuable because of their focus on the forces and problems that lead so many serious well-intentioned innovation efforts to fail. McGrath's text on product strategy for high technology companies (McGrath 2000) is very useful for its comprehensive and advanced treatment of the front end of product development. The author (Campbell 2002) has outlined an architected approach to the front end of product development, emphasizing the explicit treatment of uncertainty. Cooper and Edgett (Cooper et al. 2003) have provided important information on characteristics and qualities of individuals that are strongly related to product success. Although some of these references focus more on the process of the front end of development in general than specifically on innovation, we assert that a systematic front end process is one of the key enablers of innovation, and seek to outline an aggressive role for the *strategic system architect* in the front end.

The most fundamental distinction concerning innovation is between fundamental or discontinuous innovation and incremental innovation. Fundamental innovation establishes a new paradigm in the market that is not just an extension of an existing dominant design. Xerography provides an example. The first xerographic copiers, although they had much higher per copy cost than mimeographic machines, met dominating customers needs for ease of use and response time. They rapidly displaced competing technologies and also revealed a huge latent demand for office communication in the form of document chains. As a complementary example of incremental innovation, Xerox moved in the eighties to introduce a development technology that could simultaneously render fine lines and large solid areas, always a matter of trade-off in

earlier technology. Copy quality became far more robust in the Taguchi sense, but the core copying paradigm was not changed. In this paper we direct our approach to fundamental innovation as the more challenging case, although much of what we advocate is applicable to incremental innovation.

A second useful distinction is process innovation versus product innovation. (By "process" we mean not just the manufacturing process, but the end-to-end process (or the value chain) of building, selling, distributing, servicing, and supporting the product.) Dell Computer's success is an example of process innovation. Dell combined new technology in selling, configuring, building, and distributing personal computers to in effect micro-segment the PC market and let customers get exactly what they wanted while operating at a lower cost structure than their competitors. Wal-Mart Stores' integration of sales and customer information with a highly responsive supply chain enabling a very high inventory turns ratio is a second example of innovation across the entities of the value chain. What customers wanted was in the store at low prices.

Product innovation, in contrast, focuses on innovation in the product itself. A classic example of product innovation is the Apple Macintosh. Based on the technology inventions from Xerox PARC, Apple, on its second try at marketplace, innovated by aligning the graphical user interface technology with customer needs for an easy to use interface at the right range of price points.

Although the product/process innovation distinction is useful, it should not be viewed as a dichotomy. Some innovations cut across product and process. The fly-by-wire technology introduced by Airbus was product technology, in that it introduced a new kind of control technology into the product, but much of its benefit to the customer was realized in support areas, such as less training to fly new airplanes or move from one type to the other because of control consistency across the fleet. The xerographic example is a second example of the interaction and indeed, the compound effect, of joint product and process innovation. The product innovation inherent in the xerographic process was complemented by the value chain innovation of marketing relatively expensive xerographic machines by selling on a cost per copy basis. The penetration rate would have been far lower without the value chain innovation.

The clear implication is that the platform architecture must address both the product architecture **and** the process (value chain!) architecture. We shall use the term *system platform* when we mean the union of these, the term *product platform* when we mean the platform subset or elements underlying the product and the term *process platform* when we mean the platform subset or elements underlying the end-to-end value chain.

The Challenge. Innovating is challenging. In a comprehensive paper addressing the success rates of *substantially new commercial products*, (Stevens et al. 1997) concluded that success rates were low indeed. This paper integrated data from a number of sources, including venture capital. Characterizing success of this type of product as a surrogate for innovation success, their results demonstrate that: measured from the time of launch, success occurs 59% of the time; measured from the time of commitment to major development, success occurs 25% of the time; and measured from the time of commitment to significant development, success occurs only 11% of the time. In other terms, of nine significant developments undertaken, only one is a commercial success. Recent history of both small dot com type companies and large technology companies in innovation would seem to imply that success rates of these magnitudes are best taken as an upper bound. Accordingly a realistic goal for any organization, and indeed for the architectural-centric approach advocated in the next section, might be to halve the failure rate as

referenced to the organization's specific industrial sector.

Because innovation implies matching customer needs with technology, it worth looking at one key study on the impacts of functional disconnects between marketing and technology. (Souder 1988) found a most remarkable dependence of success on organizational "harmony" between marketing and R&D. He classified 289 "new product development innovation projects" from 53 firms in terms of whether they reflected harmony, mild disharmony, and severe disharmony between these two functions. Table 1 shows success rates in terms of these categories.

Table 1

Success and failure as a function of the quality of the working relationship between marketing and R&D.

Result	Success	Partial success	Failure
Harmony (41%)	52%	35%	13%
Mild disharmony (21%)	32%	45%	23%
Severe disharmony (39%)	11%	21%	68%

The impact is striking. The success rate is more than quadrupled by good working relationships compared with severe disharmony. Even more strikingly, the failure rate is lowered by a factor of over five. (One might well ask at this stage what the impact would be of strong working relationships between all functional groups involved in product development and in the front end in particular.) The implication for the system architect is clear – the platform work must support and indeed enhance good working relations between these functions. These data (and other similar data) strongly that a concurrent approach to innovation is much more likely to succeed than a conventional serial approach.

To focus our analysis, we now proceed to address the three key questions that arise in the early stages of any innovation thrust.

1. Which customers and what specific needs?
2. What technology sets offering what specific benefits?
3. How are the customer needs and technology benefits to be aligned and integrated into a set of offerings?

We address these questions in the next section at the total system level, that is, the system incorporating the target market, the development chain, the delivery chain, and the competition. We treat first the function of the system platform as the central structure aligning customers and technology in some detail, then sketch out the other functions of the platform.

Strategic System Architecting

Platform Views. For purposes of this paper, a platform is defined as *that set of assets, including the rules for their application, common to the entities of the product system.* Since our focus here is not just the product itself but the product embedded in the whole system of how it is developed and distributed and marketed, we will use the term *system platform* to mean the platform underlying this top level system. The *product platform* accordingly is a sub-platform in our formulation.

We use the term "assets" to mean not just physical assets of the product like chunks of software code or specific hardware implementations, but also designs, technology, and architectures underlying the product. On the value chain side, assets include not just physical assets like production lines and customer data bases, but technology and architecture as well.

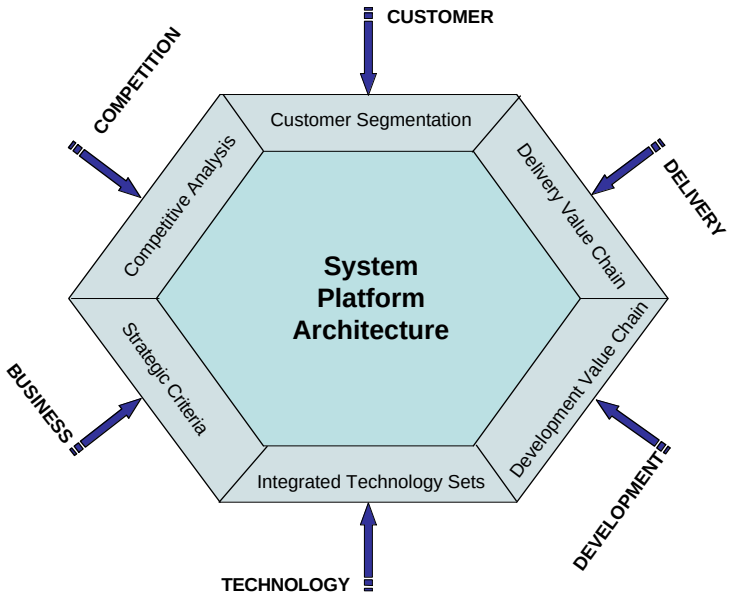
Most important, perhaps, is the knowledge gained in the relevant customer/market and technology areas as this knowledge has high leverage in times of rapid change when specific implementations and designs are rapidly obsoleted.

At the early stages of any platform development, much of the content of the platform will be intellectual or intangible assets, transitioning to more physical assets as the platform is developed. There may be substantial reuse, of course, but often too much emphasis is put on the reuse of physical assets at the expense of intellectual assets.

We emphasize the breadth of our definition as opposed to the more limited definition of platform as a product platform in much of the literature for two reasons: first, to leave no wellspring of innovation excluded and second, to emphasize that success in innovation tends to be multiplicative. Everything must be done well (or at least reasonably well) across the product and the value chain to succeed; failure at one or two aspects can doom the whole effort to failure. Accordingly the strategic system architect must operate across all aspects of the system platform to support success.

We adopt a "views approach" (Kruchten 1995) to the system platform for ease in describing it. Figure 1 depicts the six views (or perspectives) of our formulation. On the north-south axis, i.e. the *innovation axis*, are the customer and technology. As the diagram shows, at the platform level customers are aggregated and segmented. The appropriate level of technology is that of integrated technology sets, as any new technology must work harmoniously with other technologies, whether new, evolving, or mature.

Figure 1. The Six Views of the System Platform Architecture



The right-facing two faces of the hexagon are the development and delivery views. We include manufacturing in the delivery process because, although technical, it is a recurring process like sales and delivery; whereas platforms and the products developed on them go through the development process only once. Accordingly under delivery, we include manufacturing, distribution, logistics, outbound marketing and sales, service, and support. Development comprises technology (set) development, platform development, and product

development. A composite view of these two faces is in effect the internal process view.

On the left-facing faces are the competition view and the business view. Competition is all important to success, and deserves to be raised to this top level. The competitive analysis and strategy are all too often the weakest link in platform proposals. The composite view across the top three faces is obviously the market view, combining the customer, competition, and the delivery view. The business view looks at the potential success of the entire effort. Costs, both investment and recurring, are based on the internal processes; revenue and share numbers come from the competitive market structure.

Another way to look at the hexagon as a structuring framework is to see how it is organized to support the three central management questions of any innovation effort:

1. What is it we propose to do?
2. How do we propose to do it?
3. Why do we expect to succeed?

A vertical cut up the center in effect addresses the first question, a vertical cut up the right side of the hex, through development and delivery, addresses the second question, and a vertical cut up the left side addresses the third question. In other words, the hexagonal structure facilitates an integrated and complete response to the three big questions.

Working at this total system level is complex. That is the key reason why a system architect is an ideal person to organize and drive the system platform work. But the strategic architect must possess business, market, and process skills fully comparable to her technical skills.

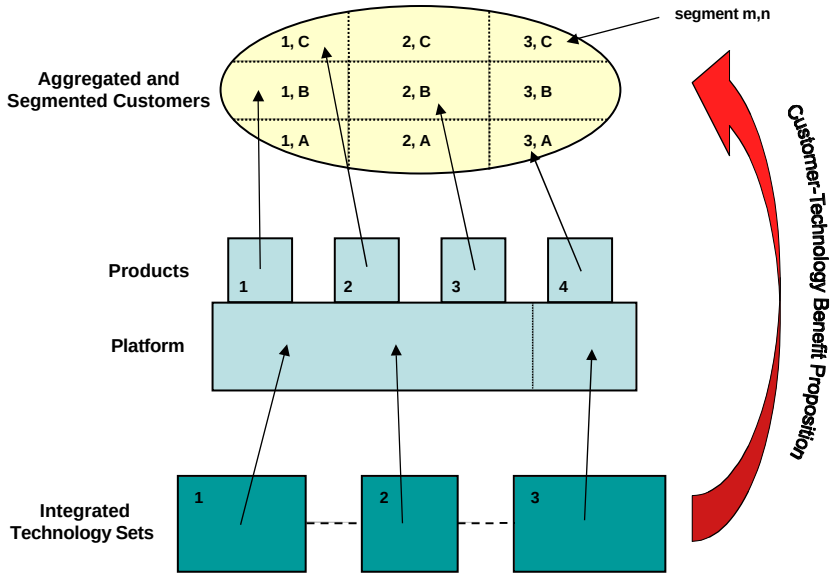
The System Platform. We look now at the platform necessary to support these six views. We assume a basic familiarity of the reader with the theory and application of the concept of platforms. Three useful references are (Meyer et al. 1997), (Robertson et al. 1998), and (McGrath 2000). McGrath's approach is closest to that in this paper because of his emphasis upon the role of technology. We focus first and most intensively on the innovation axis.

The *Customer-Platform-Technology Map* is drawn in Figure 2 to emphasize the central and dual role of the platform. From the market perspective, it provides the framework to meet the need of the aggregated customer set by appropriately combining and distinguishing their needs. From the technology perspective, it provides the framework to effectively employ and leverage the integrated technology sets to address the market. The role of the strategic system architect is clear: to synthesize the platform both to effectively meet customer needs and to efficiently leverage the new technology.

Figure 2 is of course schematic; a single platform supporting four products is depicted. The four products target a sub-set of the nine segments of the simple two-dimensional segmentation. In practice, of course, there is likely to be more than one platform, e.g. the technology sets could lead to a hardware and to a software platform, any realistic market segmentation will be more than two dimensional, and technology relationships will be more complex than those portrayed. But the principle is: the CTP Map is the core structure to a) get technology potentials aligned with customer needs and to b) get the marketing and technical functions working harmoniously.

The most fundamental innovations arise from discovering and then satisfying latent (or emerging) needs of customers. These are much harder to discover than extensions of existing needs, and call for very skilled marketing. Even when aggregated, these needs well may represent a very small initial market relative to existing markets. Yet satisfying these emergent needs is the way to establish innovation beachheads that grow into major innovations.

Figure 2. The Customer-Platform-Technology Map



Marketing and the Platform. In any corporation, there are always powerful forces, not least in sales and marketing, focussing on satisfying needs of the biggest and most profitable customers. But it is all too easy to dismiss or lose latent needs of new or initially small customers if these forces sweep all before them. Christensen's (Christensen 1997) eloquent work demonstrates how this phenomenon has caused established firms to miss the innovation boat again and again. Emerging opportunities for fundamental innovation can be easily dismissed because of projected small (initial) scale and presumed "riskiness."

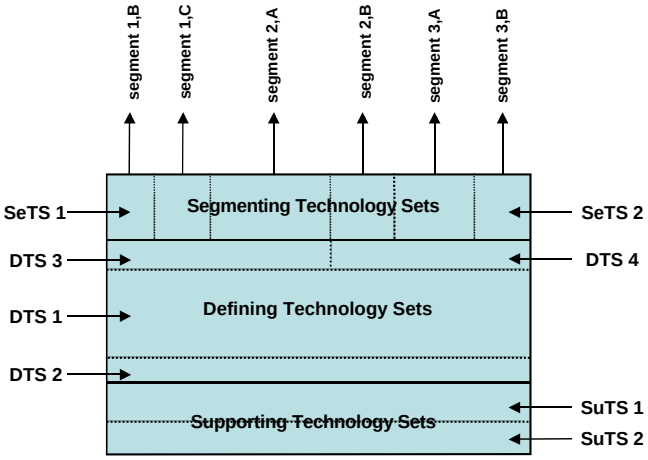
The segmentation structure is crucial here. Most often, the dimensions on which segmentation is performed are selected to do a good job of segmenting the existing marketplace for the company. Then architecting the platform to map into this existing segmentation may well limit innovation (on the first day!) For the system architect to drive innovation, she must be proactive in getting a market segmentation that effectively comprehends latent needs so they are explicitly in the CTP Map. One key to innovation can be looking at customers through a new lens, i.e. a segmentation structure that honors latent needs.

Technology and the Platform. Technology never functions in isolation (except maybe in the research lab!); a key step in its maturation is to integrate with the other major technologies, whether new also, or developing, or mature, that will be employed in the platform. Iansiti's recent work (Iansiti 1998) cogently addresses this point. The point is that robustness of the technology set can only be determined in context.

As the platform is synthesized and refined, the system architect must take a leading role in determining how the technology will play into the platform. Looking toward the customer, she should insist that the technology produce customer value and, looking toward development, that it be on a developmental track to achieve robustness. The technology must be essential to the core benefit proposition, that is the benefit to the customer expressed in the customer's words. For instance, technology can be very advanced from the technologist's perspective, but can be far ahead of the market. (Consider for instance the vast expenditures on G3 technology in

Europe.) Accordingly we advocate that a *customer-technology benefit proposition* be developed to define and express the central relationship between customer needs and technology potential. Secondly, the (projected) development track to technology maturity must be rapid enough so that platform development does not revert to a (disguised) technology development, with concomitant delays and overruns.

Figure 3. Technology and the Platform



With pressures from management to get a quick return on technology investments, from technologists to see all their work utilized, and from the field for winning products, technology selection is no mean challenge. Perhaps the most effective way for the architect is to get consensus established early on the specifics of the technology selection criteria and then hold to these criteria in subsequent discussions.

(McGrath 2000) has provided a useful definitional framework of defining technology elements, supporting technology elements, and segmenting technology elements. We think in terms of *technology sets* to emphasize technology integration. Figure 3 applies this framework to Figure 2, and depicts the mapping of technology to the platform in more detail. *Defining technology sets* are those essential to the CTBP. Four are shown in Figure 3, with two spanning the entire platform. *Segmenting technology sets* are those that provide variety and differentiation to specific products, and *supporting technology sets* are those that support the entire product family, but cannot be deemed critical to the core customer benefit. Adoption of a framework such as this can focus technology development and avoid the pursuit of interesting, but not essential technologies.

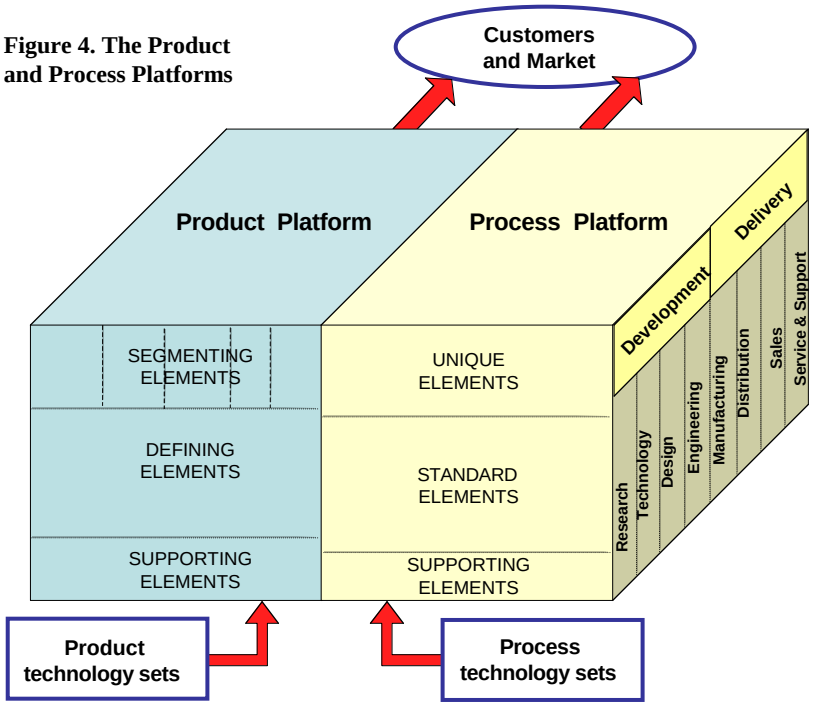
We now look briefly at the platform from the other four views.

The Development and Delivery Views. We treat these views together because we wish to emphasize the importance and value of an *end-to-end* development and delivery process (Holmes et al. 2004), that is, an integrated top level process spanning from first need or invention to market delivery and support. Using the definition of platform given above, we define the *development platform* as *the set of development assets and the rules for their application common to the projected product family*. Development assets fall into three categories: skills

and competences of individuals and organizations; development processes and methodologies; and the development environment, including tools and facilities. Just as for the product platform, the development platform is established in the front end. This is accomplished by selecting from the generally available set of assets (reuse!) and combining these with new assets where advantage can be gained. The way is opened therefore for innovation in the development platform.

Consider a specific example to fix the concept. A company lacked a competitive set of products in a low end market. Product technologies were examined, but the cost benefits were projected to be minimal, no greater than the annual average price decrease in the market, and would involve a delay. The team chartered to undertake this project created an innovative delivery platform for the company. The product platform and the product set were architected into a set of eight very well defined modules, and world class partners, from design through production, were sought for each module. Internal groups bid on these modules as well. As partnerships were built, the product/platform architecture had to be iterated twice to effectively impedance match into external capabilities. The products off the platform achieved cost benchmark, and, moreover, were developed a third faster than previous similar products. This style of development using full service development partners, sometimes called Extended Enterprise, has of course become more common in recent years, but at the time of the example, it constituted an innovation for the company.

Figure 4. The Product and Process Platforms



By analogy, the *delivery platform* is defined as *the set of delivery assets and the rules for their application common to the projected product family*. It is organized as supporting, defining, and unique platform levels, the unique level containing those elements of the value chain which are differentiated from the standard corporate ones. Examples were presented above of how companies employed new technology to innovate and achieve sustained competitive advantage

in the delivery process. The seamless combination of the development platform and the delivery platform we term the *process platform*. (See Figure 4.) This is the end-to-end value chain that will be employed for the projected attack on the market. As the product platform determines *what* will be done, the process platform determines *how* it will be done. Co-architecting these two platforms (from the first day!) develops an integrated system architecture to attack the market and to give potential innovations every chance to succeed.

The Competitive View. In the author's over 40 years of experience working primarily in the front end of product and platform development, the greatest weakness in new product/platform proposals has consistently been the area of dealing with competition. Competitive analysis is done, often on a functional basis, but it is not integrated at the strategic platform level and its implications are not fully addressed.

The central issue is to differentiate the products/services off the platform in the eyes of the customer. Define the *vector of differentiation* as the consistent basis of competitive advantage, in the words of the customer, over the segments of the target market. Four key points are:

1. It is the customer who must understand the basis of differentiation. Therefore the vector must be easily communicable to the market/customers.
2. The vector must provide a compelling reason to purchase relative to the competition.
3. The vector is strategic, providing long term advantage, over the expected platform life. Consequently it must be derived from the platform, not by short term features.
4. While vectors of differentiation are commonly sought from product technology, development or delivery process technology can be the source of advantage, especially for mature or commoditized markets.

As a technology and its marketplace mature, in particular as a dominant design paradigm becomes established, the basis of differentiation may indeed shift from innovation to cost and support. Differentiation needs be conducted across the segmented market, and accordingly the selection of a vector may explicitly select or deselect segments within the market. But the bottom line is this: the system platform architecture and its evolution should be relentlessly driven by the pursuit of sustainable competitive advantage.

The Business View. Any innovation effort requires business justification. The starting point is alignment with corporate strategic intent. Then justification for the proposed investment must be developed, generally in the context of portfolio management. For innovation efforts to succeed, the portfolio management structure must not be dominated by short term return metrics. The four crucial items in the business justification are the cost structure, both investment and recurring, the revenue and profit projections, the financial analysis tools, and the process to develop the justification.

We call the justification the *Economic Case*. This term is chose deliberately to draw a distinction with business cases. An economic case is strategic in nature and less detailed than a business case, covers the entire platform/product family, highlights the critical factors that influence the outcome, and preferably is based on options analysis style thinking, especially in the early stages, instead of net-present-value calculations. The use of options type thinking is essential to support highly innovational efforts, because it comprehends and even demonstrates how to exploit the high uncertainty that prevails in the front end.

As to process, a concurrent process is strongly recommended. In the classic serial process, the front end team does a great deal of work, the finance function is called in, the results are below corporate guidelines, and with review deadlines looming, all assumptions are looked at through rose-colored glasses until the projected results exceed corporate requirements. This

process is classic problem detection and correction(!), as opposed to problem prevention. The better approach is to start on the first day with a reverse economic case, so that *critical economic success factors* can be addressed along the way by the entire team. Reasoned and inspired decisions can be taken on how to improve these success factors, instead of following the last minute "optimism-rules" approach.

Regarding costs, the system platform approach as outlined enables the preparation of life-cycle cost models. These models in turn enable crucial strategic questions like, "What would be the results if we moved to a five-sigma product instead of a four-sigma product?" to be addressed with insight.

The platform references treat in some detail ways to justify platform investments. The issue of course is higher up-front costs because of the platform approach with lower per product downstream costs. Credibility is crucial. In the author's experience, however, the platform approach has two major benefits which are hard to quantify and can be overlooked. Even for the first product to be developed off the platform, the platform structuring and architectural and definitional efforts constitute a determined problem avoidance approach. Architectural and conceptual problems are well-known to be the most difficult to fix downstream.

The second major benefit is that the platform approach, once the initial platform is available, constitutes a very market responsive approach. With high uncertainty in customer needs, in competitive responses, in the progress of technology, and in government regulation, the ability to generate variants rapidly is a powerful competitive advantage. The time of greatest learning is when the initial product reaches the market. The platform approach enables rapid response to this learning, and can therefore reduce market risk. The potential scope of the platform must therefore cover a broad range of customers. In architecting for innovation therefore, platform approaches that are robust in the face of uncertainty are to be preferred to optimizing narrowly on the presumed customer.

Reliable revenue, and of course, share, margin, and profit projections are always challenging to obtain. To deal with uncertainty, the best approach is to develop ranges. For revenue, for instance, high, medium, and low projections are developed based on the values of the critical economic success factors. These ranges establish the degree of uncertainty, and support options style thinking.

The platform structure as outlined establishes a principled way to make these projections. For specificity, we describe a specific revenue projection approach that works for many markets. Start with Figure 2. The extent of the marketplace oval denotes the *target market*. High, medium, and low revenue projections for the target market are established as a function of time, with the underlying factors and assumptions clearly delineated. Coverage of this market grows over time, as the products and variants in the various segments are developed. Multiplying the target market size by the coverage factor yields the *served market*. Next there is the question of presence, the combination of awareness and consideration. The presence factor is a critical parameter of the delivery platform, accordingly the next step is to multiply the served market size by the presence factor to generate the *contended market*, that is the market in which the planned products will reach the finals in terms of buying decision. The final step is to multiply the contended market size by the win rate, as derived from the vector of differentiation, to generate the projected revenue as a function of time. The competitive view provides the necessary information by segment. Ideally the whole process is done by segment to avoid excess generality. In effect, we have established a *revenue architecture*.

Following an analogous process lets a complete economic architecture be constructed,

covering all top level business and financial matters, and organized as the *economic platform*, the third components of the system platform. The close integration of this platform with the product and process platforms permits all critical success factors, and their inter-relations, to be developed concurrently. Starting from day one, a complete and concurrent system platform evolves.

The Platform Development Process

Ideas for innovation are common in most corporations, coming from researchers, technologists, marketing people, etc. Many of these typically are for incremental innovation, which the normal technical management system tends to handle relatively well. Ideas that have the potential for disruptive or fundamental innovation have a much more difficult time moving forward, as pointed out so clearly by (Christensen 97.)

Ideas and concepts for a breakthrough innovatory product or service need early cross-functional input to gain robustness and get traction. A small, certainly less than eight person, core innovation team should be assembled. The core team should consist of one to two people with strategic marketing competence, one to two people with knowledge in the relevant technologies, one to two people with value chain capability as relevant, a person with strong business sense, and the system architect. The system architect, if she is not narrowly technical or junior, is ideally placed because of her broad perspective, to lead the team.

To be avoided are teams that are just sounding boards for one person and teams that lack strong knowledge across relevant domains. And if there is never any contention across the team, it is sign that a robust approach may not be emerging. The most valuable output of exploratory efforts is knowledge, specifically, the shared knowledge of the core team confident in their ability to move the concept toward the market and success. The organizational corollary of Gresham's Law is that the urgent drives out the important, so it is far better for the team to work in short intensive and interactive bursts than to have the strategic work as everyone's n-th priority. These bursts are separated by periods in which technology development and customer understanding are pursued.

Our recommendation is that a short and intensive platform architecting effort be devoted to promising ideas for innovation using the core team approach. The early stage of innovation must be fairly fluid and flexible, therefore the *co-evolution process* shown in Figure 5 should be taken as a framework or guideline process, not as a rigid process for "innovation management." The team should be careful not to rush to conclusion; carrying a number of options and alternatives forward is an excellent way to gain robustness.

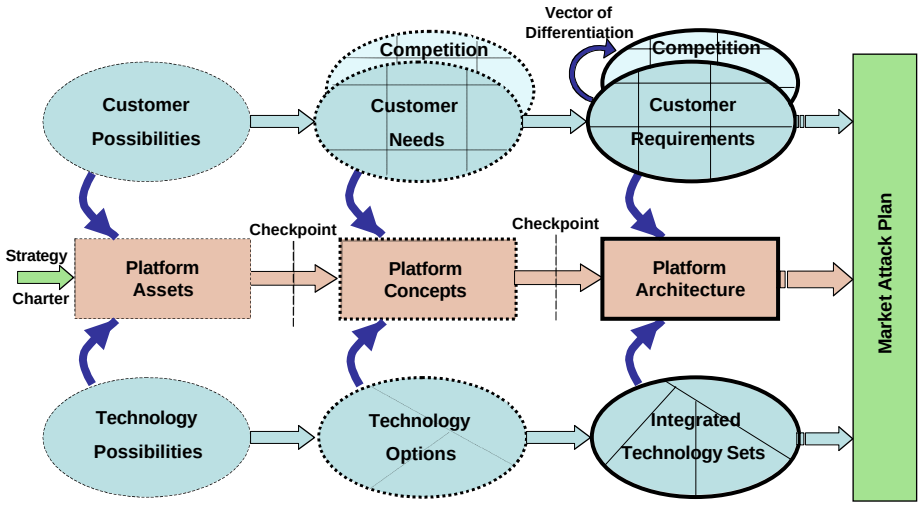
A concurrent process, with three iterations, is depicted. Shorter and more frequent iterations are likely to be preferred in practice. The concurrency of the process is crucial. Serial processes in which marketing develops requirements, planning turns them into specifications, and engineering develops a technical response are far too sluggish for many markets. More importantly, serialization is very unlikely to yield the robust type of innovation enabled by a competent and well structured multifunctional team working concurrently.

The process starts with rough notions of customer possibilities at least loosely aligned with the technology possibilities. Note the technology possibilities include both product and process technology. It is driven by the corporate strategy and charter of the performing unit. The unifying structure is the system platform, as above the union of the product platform and the value chain platform. First concepts of the platform emerge concurrently as technology possibilities are converted into realizable technology options and as more knowledge of customer needs is

developed. Although the system architect should be the platform champion, the architecture is so fundamental to this approach that it needs shared ownership by the entire team to support development of shared knowledge.

The core of the marketing approach is to group and segment customers into subsets with

Figure 5. Concurrent System Platform Development Process



related needs. This forces more care in customer understanding, and also helps avoid the problem of trying to do everything for everyone. A evolving segmentation is depicted, two dimensional for simplicity. The range of customer needs addressed is important. The idea, of course, is to design the platform to enable a competitive product in each segment with minimum incremental development cost and time. The sequence of engaging with the various segments must be determined, e.g., start low and move up in capability as the technology develops. Furthermore, it is often found that the platform cannot extend effectively across the entire targeted customer aggregation, so that selection of a subset of the segments is made, resulting in the served market.

On the technology side, technology possibilities become refined into more specific technology options and then into integrated technology sets. Major innovations very frequently depend on the integrated maturation of more than one technology. It is important to recognize this early on so technology interaction or unreadiness problems do not prove to be the Achilles heel of later development efforts.

The end-to-end process platform co-evolves with the product platform. As discussed above, the central question to be addressed is whether the corporate standard process platform will suffice, or if innovation is mandated here.

The process results in a Market Attack Plan. The aggressive language is deliberate. An (emerging) market is to be attacked. The major components of the plan are threefold: (1) the plan for maintaining continuing closeness and relationships with the targeted customers to avoid surprise and to be able to react rapidly to change, (2) the technology readiness plan to ensure the maturation of the selected technology on schedule, and (3) the plan for developing the system platform itself, including maintaining its economic justification.

Two checkpoints are shown. These should be lightweight checkpoints in that they do not get

into needless detail, but keep attention on the major customer, platform, and technology issues. They provide opportunities to redirect the entire effort or possibly to cancel it in favor of better opportunities. The focus should be a learning focus. What have we learned since the last review to support and validate moving ahead with this opportunity?

The Role of the System Architect

Summary. Much of the role of the strategic system architect, and the requisite multi-functional skills and competences, can be inferred from the above material. We highlight the special requirements to function at the strategic level. Within the team, the central role of the architect is to lead the conceptualization and development of the platform; outside the team, the central role is to use the platform framework for structured interactions with all stakeholders, especially business sponsors. The system architect should certainly be qualified to lead the multifunctional team as described, but equally well could act as a strong number two. The advantage of the latter is that she could devote more time specifically to the architecture, but the potential concern would be the tendency for the architecture to devolve to a purely technical framework.

The approach described is a highly concurrent, integrate-and-win, approach. But the role of the system architect is not to perform the integration herself, but rather to use the platform construct as the vehicle for the core team members to work together. The platform functions as the structure for team or shared knowledge development.

By contrast, in system engineering, interfaces are structured and maintained by the system engineer (divide-and-conquer approach) so that large groups can work relatively independently in their own domains. Systems engineering will be required downstream, before the development effort is ramped up, but it must not be applied too early, as the organic functioning of the team will be limited. It is all to easy for individual experts to return to their functional foxholes.

Figure 6. Integrative Role of the System Architect

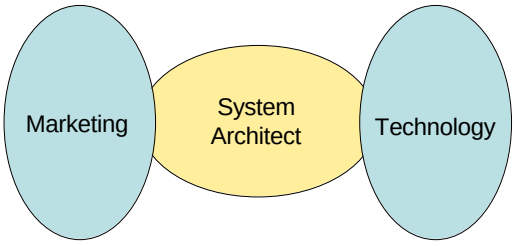


Figure 6a. System architect in interface role.

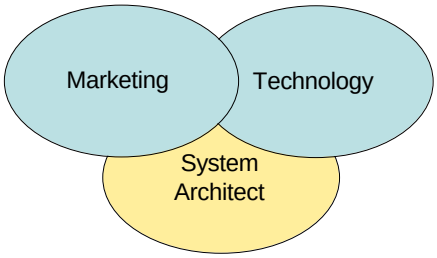


Figure 6b. System architect in integrative role.

Accordingly the strategic platform architecture must be employed not to put barriers between

the experts gathered for the core multifunctional team, but facilitate close interactions, dialogue, and shared decisions. Figure 6 has been prepared to show how the architect, and by implication, the platform should function in the case of technology and marketing expertise. At the top (Figure 6a) the interface has been engineered. Marketing and technology are separated with an interface between them. The pressure then is for the interface to become well documented, i.e. for the technologist/engineer to demand "the requirements," which then are put into a requirements management system and controlled from there on. Too early systems engineering of this type serializes the process, loses concurrency, and almost certainly will place a huge damper on the kind of creativity and exploration need to support innovation.

On the bottom (Figure 6b), the system architect is shown in the integrative role. The system platform is the central object underlying the integration, a shared knowledge structure, spanning the domain boundaries. The architect facilitates and enables high quality interaction, with the platform serving as the language of the dialogue. She can focus the dialogue by establishing the set of methods that operate on the platform object. Further, she is also positioned to bring her personal energy and creativity to the dialogue.

It is fair to say that the approach outlined is the paper is complex. But it is well demonstrated that there is no silver bullet of innovation. As system architecturing may be said to be the practice and science of complexity, the strategic system architect is the ideal person to realize value from this system platform approach.

With specific respect to innovation, the system architect herself must be sensitive and responsive to new ideas. It is not too much to say that she must cherish new ideas as they emerge, and give them the opportunity to find a home in the platform so they can have a chance to mature and grow. At the initiation of a platform effort as outlined herein, many new ideas will come forward and will be generated at the team proceeds. These provide a rich background of creativity to fuel the platform activity. Most importantly, the strategic system platform approach affords a way for the system architect to directly counter the concept of the "fuzzy front end." Too much potential innovation has been halted in its tracks by presumed riskiness. Of course, uncertainty is always present, but the strategic platform concept has the robustness to bridge the worlds of corporate business management and of the forces driving innovation.

Conclusions

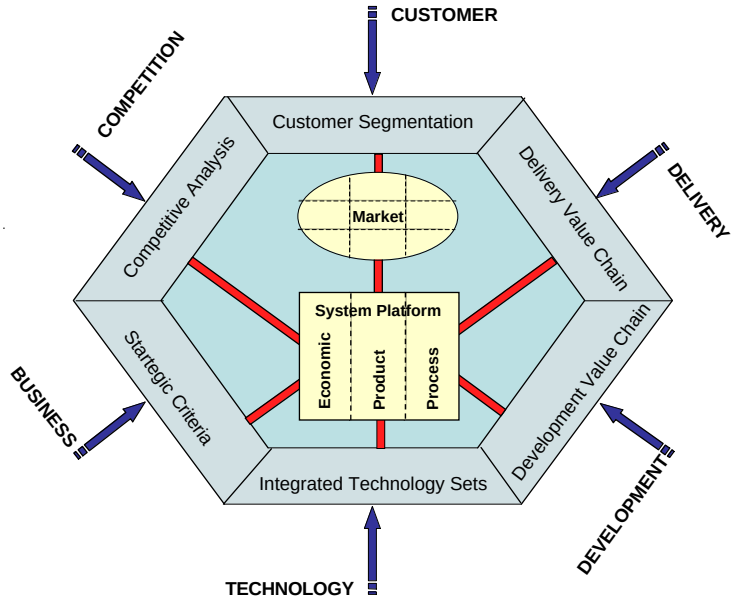
Rechtin (Rechtin 91) observed that much of the value of an architecture comes from the relations between the entities. So it is in this case. Figure 1 has been redrawn as Figure 7 to highlight the key relationships (depicted in red) in the hex and to show a further level of detail as developed in this paper. The system platform is shown in the center, connected to the customer via the market and to the wellspring of technology. The system platform is comprised of the three sub-platforms addressing the three central questions: the product platform – what we will do, the process platform – how we will do it, and the economic platform – why we will succeed.

In conclusion, we summarize the three central points of the approach.

1. **Concurrent.** The system platform approach is concurrent from day one. The (concurrent) whole is greater than the (serialized) sum of the entities.
2. **Complete**. The approach is complete in its scope. This directly addresses the multiplicative nature of market success, but more importantly, permits all potential wellsprings of innovation to be comprehended for integration into the platform.

3. **Innovation Enhancing.** The system platform approach provides a vehicle to move nascent ideas and concepts for innovation forward in the challenge of the corporate environment. The use of a multifunctional team from day one to build shared knowledge, confidence, and commitment is key. The use of options-style thinking in the context of the complete platform deals effectively with uncertainty and risk perceptions. And finally, the integrated and continuing focus on the customer, competition, and the market throughout the front end cannot but help meeting real needs and enhancing the likelihood of success.

Figure 7. The System Platform Architecture



References

Campbell, R.B., "System Architecting the Front End of Product Development." *Proceedings of the 15th International Conference on Software & Systems Engineering and their Applications* (Paris, France, December 3-5, 2002).

Clausing, Don, *Total Quality Development*. ASME Press, New York, NY, 1994.

Clausing, Don, Axelband, Elliott, and Campbell, R.B., "Advances in Commercial Product Development – Lessons for INCOSE Systems Engineering." *INCOSE Insight*, Fall 1999.

Clausing, Don and Fay, Vincent, *Effective Innovation*. ASME Press, 2004.

Christensen, Clayton M., *The Innovator's Dilemma: When New Technologies Cause Great Firms To Fail.* Harvard Business School Press, Boston, MA, 1997.

Christensen, Clayton M., *The Innovator's Solution: Creating and Sustaining Successful Growth.* Harvard Business School Press, Boston, MA, 2003.

Cooper, R.G. and Edgett, S.J., "Benchmarking Best Practices in New Product Development in NPD: Part 1 – culture, climate, teams, and the role of senior management." *Research-Technology Management*, 2003.

Iansiti, Marco, *Technology Integration: Making Critical Choices in a Dynamic World*. Harvard Business School Press, Boston, MA, 1998.

Holmes, Maurice and Campbell, Ronald B. Jr., "Product Development Processes: Three Vectors of Improvement." *Research-Technology Management*, to be published, 2004.

Kruchten, Philippe B., "The 4+1 Model View of Architecture." *IEEE Software,* Vol. 12, No. 6, November 1995.

McGrath, Michael E., *Product Strategy for High Technology Companies: Accelerating Your Company to Web Speed.* Second Edition, McGraw-Hill, New York, NY, 2000.

Meyer, Mark H. and Lehnerd, Alvin P., *The Power of Product Platforms.* The Free Press, 1997.

Rechtin, Eberhardt, *Systems Architecting – Creating and Building Complex Systems*, p. 48. Prentice Hall P T R, Englewood Cliffs, NJ, 1991.

Robertson, David and Ulrich, Karl, "Planning for Product Platforms." *Sloan Management Review*, Vol. 39, No. 4, Summer 1998.

Souder, W. E., "Managing Relations Between R&D and Marketing in New Product Development." *Journal of Product Innovation Management*, Vol. 5, No.1, March 1988.

Stevens, Greg A. and Burley, James, "3000 Raw Ideas = 1 Commercial Success." *Research-Technology Management*, Vol. 40, No. 3, May-June 1997.

Utterback, James M., *Mastering the Dynamics of Innovation,* Harvard Business School Press, Boston, MA, 1996.

Biography

Ronald B. Campbell, Jr. is currently a Senior Lecturer at the Centre for Innovation in Product Development at Massachusetts Institute for Technology. He consults for high technology companies in the area of product development business processes with the Business Process Systems Group. He was at Xerox for 17 years in management roles responsible for the development of technology, architecture, and new products and served as Chief Technical Officer. Prior to that, he was at Raytheon for 18 years with responsibilities for advanced development and research and served as Vice President for Research and Development. Campbell has a PhD from Harvard in Applied Physics.