Make platform innovation drive enterprise growth
Marc H Meyer; Paul C Mugge
Research Technology Management; Jan/Feb 2001; 44, 1; ABI/INFORM Global
pg. 25

MAKE PLATFORM INNOVATION DRIVE ENTERPRISE GROWTH

A robust platform strategy can be a winning business strategy for R&D as well as the entire company.

Marc H. Meyer and Paul C. Mugge

OVERVIEW: Most corporations lack a precise, operational definition of their product platforms, without which progress is difficult to achieve. With that definition, new product strategy must be recast to capture the degree to which common architectures and subsystems will be leveraged across existing and new market applications, as well as the timing of product introductions. Managers must also reconsider the financial model of the business, including investments in the development of platforms versus products, and the changes that are possible in the margins and contribution of platform-based products to profits. Perhaps most pressing are the organizational questions: How can the R&D executive best structure the organization and the processes by which groups plan and interact so as to facilitate platform development and platform-based product development? Recent efforts of IBM and other corporations to embrace platforms provide both a management and a technology paradigm.

Product platforms are an increasingly well known engineering concept, with applications well documented in fields ranging from automobiles to computers, software, medical equipment, and even services (1–4). However, our research suggests that platforms are much less understood, let alone popular, as a business concept. This is unfortunate because we believe that platforms are a common-sense way for a firm to leverage technology into new markets and, at the same time, reduce per-unit costs through more efficient production and procurement. Well-managed platforms allow companies like Hewlett-Packard, IBM, SUN, Cisco, and EMC to grow revenues while reducing per-unit costs.

One would think that providing the opportunity to increase revenues while improving gross margins would claim the instant attention of any executive. However, this does not seem to be the case. Proving the business case for platforms takes work, albeit a different kind of work than designing next-generation products, services or processes. Because the creation and renewal of product platforms is the essence of business success in so many different industries, and because new platforms
require some degree of new technology, we believe the senior R&D/technology leader is the logical choice for making that case. It will take fortitude and persistence, and the willingness to challenge conventional processes and organization. However, in the long run, a robust platform strategy is a winning business strategy, both for the R&D function and the entire company.

This article is focused on helping you, the R&D leader, to make that business case. We shall describe how other successful companies have used platform management to competitive advantage. In particular, we will profile the results achieved by IBM’s hardware businesses. External cases such as these can become useful elements in your arguments to renew technology and product lines. Then, we provide a simple approach for you to make your own business case that focuses on helping your colleagues understand and develop answers to two basic questions:

- What is a product or service platform for your business, and how does the platform evolve over time?
- How do robust platforms—and the innovation in product, process and services embodied in those platforms—drive business growth?

Defining Platforms for Your Business

It is essential to gain organizational consensus on the definition of platforms for your business and, if possible, to facilitate the terms and language by which various groups define their own product platforms. This facilitates component and process reuse. If every group comes up with its own definition, the consistent reuse of common parts and pieces will remain a dream.

How might you go about defining your platforms and how they can be leveraged into different target markets?

The first rule that we have learned through experience is that a simpler approach to platform definition is invariably better. Indeed, the purpose of a management concept is to simplify complexity so that management can more effectively grasp new ideas and make better decisions. For platforms, this means simple pictures of major subsystems and interfaces between these subsystems. We refer to these as “block diagrams.” It also means simple descriptions of the purpose and technologies within subsystems and interfaces. A clear articulation about how the common subsystems can be leveraged into different product groups is also essential.

Figure 1 shows a generic representation of a product platform and how it can be leveraged in the form of specific products or services into different target markets. The platform is composed of basic subsystems and interfaces. We portray market applications as a market segmentation grid, where new markets are as important to identify as existing markets. Different levels of performance and price, company size or customer age are common separators to create tiers of niches within target markets.

As we have studied companies across industry, we have found the simple picture in Figure 1 to be quite powerful. It helps management identify the types of common technology and processes shared across products. Common architecture and shared components reduce cost of goods and improve time to market. The market segmentation grid also highlights new product initiatives that enhance revenues by applying corporate assets to growth markets.

Development teams regularly ask, “How many subsystems should we show in our platform diagrams? How detailed should we get?” To answer that question, you must determine for yourself the uses of a platform diagram. One use is to communicate common architecture to different engineering groups so that they can build specific products that share common subsystems and interfaces. Another use is to communicate with people inside the company who are not engineers—e.g., marketers and finance people. The delineation of subsystems can also serve as the basis for organizing specific user requirements and the approaches and technologies used by competitors to solve those requirements.

For all these reasons, if the platform architecture is too detailed, it loses utility. On the other hand, if the architecture has too little granularity, it cannot be used to rationally organize user requirements and alternative technical approaches. Each company must find that balance through the work and judgment of its product teams.

It also helps to see how other corporations have defined their product platforms. Figure 2 shows a high-level framework for categorizing the universe of products and services for which we have studied the definition and application of platforms. The two axes are the degree of assembly (from assembled to nonassembled) and the
degree to which a product is physical or intangible. It is also interesting to note that most academic research in the management of technology has focused on assembled physical products, which is just one region of our grid. Nonassembled products—nonphysical products such as software and services—have received little systematic attention, even though they represent the growth areas of our economy.

Figures 3, 4, 5, and 6 show the actual platform diagrams of products and services that have been used by teams for platform design and subsequent product or service development. We can quickly see the principles of subsystem innovation, solving latent needs and subsystem reuse in these cases. Figure 3 illustrates the platform architecture of an assembled product, a diaper. Note those subsystems that comprise “the core” and those subsystems that comprise the “attachments.” Each subsystem can be the focus of innovation over successive product generations, be it rash reduction or better attachment systems. Similarly, each subsystem can be leveraged across different product lines to achieve lower costs, such as a common core for newborns, infants and adults.

A common product architecture, shared subsystems and standardized interfaces such as those portrayed in these examples need to be agreed to across product line development groups before any further platform development or product design can proceed. In fact, most corporations find that building toward a common platform is fairly straightforward once that platform is defined. But getting to that definition can be difficult for all sorts of reasons, many of which are organizational and political.

Figure 4 shows both the platform diagram and the market applications of a nonassembled product—a material used in woven and non-woven finished products. We refer to such diagrams as “platform-market maps,” and they can serve as the essence of a firm’s new product strategy, when management decides to take platforms seriously. At the bottom of this manufacturer’s platform-market map is a high-level representation of a new process platform that can be adjusted to create different deniers for a variety of market applications. At the top are the various market segments and applications within them. On that grid are lines indicating the older manufacturing processes maintained by the company (Spin 1 and Spin 2). The new platform takes the company into new market applications, including those markets offering very high growth. Use of the process platform across markets reduces capital costs and simplifies plant maintenance. The new process also reduces emissions and improves various safety problems in the older technology.

The software firm can also use platforms to good advantage. First, it can create different applications for
different uses or markets based on common programs, data and processes. Second, it can create tools for other developers to build “plug-in” modules for its own platforms. In this way, the software company can become a channel for other developers, using their resources to build an ever-larger product family. This strategy lies behind the success of software industry leaders that include Microsoft and Oracle. The modularity of software provides great opportunities for strong platform architectures.

Figure 5 shows the platform-market map for a software product family of real-time embedded system development tools. Users of this company’s products build real-time applications using the firm’s tools and then download these programs into read-only memory chips as part of closed, special purpose systems. The definition of this company’s platform has evolved from providing an entire operating system (a real-time version of Unix), to just one specific subsystem (a real-time scheduler) that works seamlessly with different operating systems (Unix and Windows and now, Linux). Equally important is that other software companies make “plug-in” additions for this firm’s real-time toolkit and its specific vertical markets, and in some cases sell their wares through the company to reach these markets (5).

Lastly, Figure 6 shows the platform strategy of a services company—in this case, a worker’s compensation insurer. This illustration shows the integration of competencies, platforms and markets for a given company. The design driver behind this company’s services has been to reach into its customers’ business to first understand, and then help solve, their respective business problems. While other insurers focused primarily on claims man-

---

**Figure 3.** This is how a diaper manufacturer defined a common platform architecture for its different product lines.
management, this innovative company broadened its value proposition to the larger problem of preventing injuries from occurring in the first place. Then, when injuries did occur, the company developed processes to get workers treated appropriately and back to work healthy. It developed specific programs to lessen workplace hazards and a national network of physicians and therapists expert in occupational injuries, and built these capabilities into its insurance contracts. Measurements for quality and cost have been developed for each platform subsystem, e.g., the effectiveness and efficiency of injury prevention, injury reporting, medical treatment, and claims processing. The results have been outstanding, with claims incidence and dollar losses far below the industry averages.

We find that understanding services innovation is very important, even for traditional product makers. Suppliers of complex systems, for example, find that they must provide standardized, professional services to their customers to ensure successful implementation and integration. Software companies are increasingly positioning themselves as service businesses through Web hosting applications, systems integration or custom programming to augment standard products.

Principles of Platform Management

The cases we have described reflect basic principles of platform management:

- **Platform renewal must be continuous. Innovation occurs at both the level of the overall architecture as well as the finer level of the individual subsystems and interfaces.** Product, system and service architectures do not necessarily have to be completely discarded to create next-generation products, unless, of course, the architecture itself is so old that the consensus within the company is that “it’s got to go!” At such times, executives must be prepared—no matter how hard it may seem—to bulldoze everything into the proverbial parking lot: old product designs, old materials and components, and old plant and equipment. At other times, there is no common product architecture. Each product is unique. Platform projects can be used to create a common architecture that is necessary to start realizing the efficiencies of common parts and pieces. Enhancements to one subsystem accrue to the entire family of products that share this subsystem—by design! Such subsystem enhancements extend the life of the product architecture.

- **Interfaces can be incredibly important.** Structured interfaces allow the assembled product manufacturer to share components across many product lines, thereby reducing materials costs. For software firms, interfaces allow one to better isolate problems when they do occur, as well as to provide a clear avenue for new applications development. Clear interface design allows the single customer to navigate through a complex service organization without getting lost and frustrated, be it a bank or
a hospital. Indeed, many engineers would say that interfaces are so important that they define the architecture of a product or service family more than individual subsystems. When we ask teams to create the block diagrams described above, we have them make two versions. The first is the conventional form shown in the diaper example, where the subsystems dominate. The second is an inverse perspective, where the interfaces between subsystems are greatly enlarged, described in detail, and approaches and standards shown for each interface.

- **Clever architects design their platforms to enable recurring revenue.** These may be the plug-in modules found in software or the accessories provided by manufacturers of office furniture, power tools or automobiles to accompany their base products. Systems manufacturers in industrial markets, for example, relish the opportunity to upgrade their devices with new application software and networking cards. In services, the essence of recurring revenue is the “up-sell,” where an Amazon.com first builds its affinity group on book purchases and then expands its offerings into other related products.

- **Manufacturing should not be forgotten as a key enabler.** Gillette fixed the geometries and interfaces for its blades so that one set of blades could be produced in tremendous volume for the male and female razors. Black and Decker fixed the circular diameter of its power tool motors and achieved greater power by simply expanding the length and wrapping more copper around the motor field. Indeed, if we look at manufacturing in its broadest sense, the high-volume producers across many industries have used standardization in subsystems and interfaces to drive production and deployment costs down.
Growth comes by attacking new markets, leveraging existing competencies into new platform architectures for new product or service lines. Two companies frequently noted for their high market capitalization are General Electric and Microsoft. Although in different businesses, each has created platforms that leverage technologies and competencies into new markets. GE, for example, has developed a tremendous business in financial services such as consumer credit, insurance, and car and truck leasing. These are indeed new markets when compared to the consumer electronics and aircraft engine businesses for which GE was known 20 years ago. Microsoft has similarly applied its platform technologies to word processors, spreadsheets, databases, networked communications, and many other applications, moving from the desktop PC, to the networked PC, and recently to the handheld PDA.

There is another important subtlety regarding platforms. The cases above feature single-platform architectures, implemented with current-generation subsystems and interfaces, that serve as the foundation for derivative products or services. This approach can be very powerful within business units for carefully defining the value added for each new product and reducing costs and cycle times for each variation. Reuse of common subsystems between business units can amplify cost and time-to-market benefits. To achieve commonality between product lines, however, you must look at major subsystems themselves as strategic
platforms. Deploying common subsystems and interfaces as Lego-like platforms is not a new or radical concept. Software companies have developed and shared device drivers or libraries with programming objects for years; car companies share chassis. Gillette shares razors and Black and Decker shares motors.

To develop a useful definition of platforms in your company, we suggest that you adopt both perspectives. First, take the systems perspective and have each logical business unit define its platform architectures, how they are likely to change, and what markets they can serve. Then, get the groups together and have them see which subsystems—parts, materials, processes, software modules, etc.—can logically be shared, co-developed and procured in volume to achieve savings and enhance quality. We also suggest that you conduct an internal study of successful product platforms in the past—every company has had them—to understand the types of R&D leverage, cycle time leverage and revenue generation that is reasonable in your industry for platform-managed product lines.

Building the Business Case for Commonality

Even when platforms are clearly understood, divisions within a corporation typically have their own product or service platforms and source their own components, even if these different divisions essentially do the same thing. No one within the corporation has shown the ability or fortitude to make the business case for shared parts and pieces and to drive that concept throughout the corporation.

Only the R&D executive, with the support and force of the CEO, can hope to achieve commonality and the benefits that can come with it. He or she must not only understand the future evolution of the company’s product, process or service platforms, but also show skill and foresight to create the organizational processes and working groups required to make the vision real.

IBM is a rich example of a company that has used platform thinking to reduce both its cost of goods for manufactured products and its cycle times for the development of new products. The company has leveraged the principles of sound platform design through a management initiative that is called Common Building Blocks (CBBs). This became an approach to reducing the cost of goods and speeding time to market. Additionally, IBM leveraged its platforms into new growth markets; e.g., to link common architectures to the markets that IBM wanted to serve.

In 1990, IBM reported net earnings of approximately $6 billion. A year later, it reported a small net loss. In 1992, the loss approached $7 billion, and by 1993, exceeded $8 billion. Most of us forget just how deeply the company was entrenched in old bureaucracy, traditional markets and dated architectures. Its benchmarking studies showed that IBM’s ratio of R&D expense to revenue (E/R) was 12 percent whereas its higher growth, “best in class” competitors’ ratio was about half that percentage. IBM’s competitors were twice as productive with their R&D investments. The company was also slow to market with its new products. A full third of the projects associated with new product development experienced over two times the development cycles of IBM’s best-in-class competitors, and another third one-and-half times. No projects were done in the same amount of time as its best-in-class competitors. IBM had to address this problem.

Parts reuse was almost nonexistent in IBM during the early 1990s. An internal study found that across 20 product models in IBM’s Server product lines, approximately 2,300 part numbers were used just once across all 20 models, about 750 part numbers were used twice across two models, about 500 were used three times, 50 parts were used in four models, and after that only a handful used five or more.

Figure 7 shows three Server hardware divisions within IBM, the total number of parts in each Server product line, and the number of parts shared between the respective divisions in the early 1990s. Even for those readers who have never designed or manufactured computers, this minimal degree of reuse does not make sense for products aimed at similar markets and developed within the same corporation. Yet, for many corporations, the lack of common architecture is today’s reality.

A broader internal study encompassing 1,500 hardware models showed that even greater complexity was created downstream beyond product development from this lack of commonality. These 1,500 products had over 12,000 features, 103,000 bills of materials, and 540,000 active part numbers! There were 5,000 active production suppliers. The systems used to run the business were largely older, legacy systems that could not share data readily. There were 59 different floor management
systems in the factories, 25 different material logistics systems, and 12 different purchasing systems.

These data were indicative of processes, organization, culture, and incentives not aligned to jointly develop or source common components. The development of strong platform architectures within divisions, needed for common components, were not part of any formal planning processes used in engineering or manufacturing at IBM. This remains the status of the management of product architecture in many corporations today. Most engineers speak the language of “architectures” and “platforms,” but their companies’ new product development processes lack frameworks and tools for the specification, utilization and renewal of platforms. Without this discipline, however, products and components proliferate beyond reason. The situation was so dismal that during the early 1990s, when IBM decided to implement an enterprise-wide information management system, the sheer number of part numbers and bills of materials savaged the new ERP system. Something had to be done.

Gerstner Issues a Challenge

Enter Lewis Gerstner. Himself a former customer of IBM (while at RJ Nabisco), Gerstner as CEO of IBM would not allow turf or tradition to stand in the way of common sense. His goal was to substantially simplify and streamline IBM. He challenged senior line executives, individuals responsible for running specific businesses within the corporation, to step back and reengineer IBM’s product development and manufacturing processes. Fortunately, many of these executives had been involved in the System 360 mainframe development, introduced in 1964. That, too, had been a rich, robust architecture from which variations had been created and many markets served; i.e., it had been a great platform for its time and many executives intuitively understood the discipline and benefits of platform thinking.

A dedicated hardware development process reengineering team was formed to create and help implement a new approach to product development. The team was sponsored by a senior vice president of manufacturing and technology but served all the general managers of IBM’s product divisions. Gerstner placed ultimate accountability for these new processes and their results with these line managers. This was not to be another staff assignment, like so many change programs before it.

The process reengineering team proposed that an explicit platform management discipline had to fill the apparent gap between IBM’s higher-level portfolio management process and its single-project stage-gate process. Line management approved. Additionally, a new set of performance measures was required to understand the effectiveness and efficiency of product development (often called “pipeline” management) that focused on the platform utilization dimension.

Figure 8 shows the intersection between the portfolio, platform and the pipeline management processes at IBM. The new process model addressed not only the classical stage-gate process used to guide any particular project but also the more demanding and critical process of managing a portfolio of projects. Selecting the right markets to go after in the first place is obviously critical for success. Not only must the target market have attractive demand characteristics, but also the corporation must have or rapidly develop the technological and distribution capabilities required for success. Platform management connects the firm’s strategic product architectures and its core technological capabilities to the targeted markets. Pipeline management focuses on two types of efforts: 1) the specific products driving off existing platforms—at a rapid pace, and 2) the renewal of or creation of new platforms—which takes longer, but is nonetheless essential for next-generation products.

As these processes were developed and implemented throughout the hardware divisions of IBM, it became clear that cross-functional teams were needed to integrate knowledge across functions for effective and fast decision-making. The typical hardware division has had directors for the division’s respective hardware and software product groups, as well as directors of hardware and software technology groups, a combination of product and functional organization. While some corporations have experimented with extreme forms of doing away with product and functional departments altogether, for IBM this was never an option. The close interaction between engineers has produced, and continues to produce, breakthrough innovations in many technological fields.
New Cross-Functional Teams

Senior management decided that cross-functional teams would be needed to integrate product and technology perspectives for both new product planning and program management (Figure 9).

The first of these new teams is the Portfolio Management Team. There is one Portfolio Management Team per division at IBM. It comprises the functional heads in each of the product divisions. These individuals include the vice presidents of marketing, engineering, operations, and distribution along with the CIO and CFO—i.e., the top decision-makers needed to reallocate resources to new market opportunities. The division general manager chairs the Portfolio Management Team meetings.

Reporting to the Portfolio Management Team is a specialized team called the Product Line Planning Team. Product line planners include the division’s best systems architects and business development personnel. Collectively, they plan and map the product platforms that will be used across the division’s product lines. Technology roadmaps, resource loading diagrams, and market positioning analyses are all part of the work presented by the Product Line Planning Team to the Portfolio Management Team.

Third, for each new platform, or new product based on an existing platform, management assigns individuals from marketing, engineering and operations to what is called a Product Development Team. These teams are responsible for seeing a project through to completion, establishing clear goals for schedule, functionality, base manufacturing cost, budget, and quality. Team leaders follow IBM’s pipeline management process, a discipline for market planning, budgeting and progress review. These Product Development Teams also report to the Portfolio Management Team.

To summarize, the organization structure embraced three new cross-functional teams, each with a different focus: the Portfolio Management Team (focusing on markets, competitors, investments, and revenue growth), the Product Line Management Team (focusing on new technologies, platforms and product families), and the specific Product Development Team (focusing on platform projects or specific product projects).
new management structures were installed in every product division of IBM.

The cross-functional teams described above proved effective for creating common architectures within divisions, but did nothing to ensure the sharing of sub-systems and subsystem components between divisions. Improving reuse was essential to improve the cost of goods position of the corporation. Executives at IBM formed three additional teams to build inter-divisional collaboration.

- At the highest level is the Development Executive Council, composed of R&D executives from all divisions, whose charter is to improve commonality and connectivity between the different computer and storage systems developed by IBM.
- Development/Design Councils were also formed to build common subsystem architecture and interfaces across divisions such as for the I/O subsystem or power supply subsystem that all computers and storage systems required.
- Next, these subsystems all share basic types of sub-assemblies, components and semiconductors, most of which were procured from outside vendors. Commodity Councils were formed to manage the “preferred parts” process for each of the major subsystems, such as power supplies, storage systems and memory. These Commodity Councils consist of engineering and procurement managers from the major hardware divisions. Their focus is to ensure that wherever possible, the same component is used (and therefore purchased in greater volume) in the various product lines.

A Five-Year Journey

For IBM, the initiative to redesign the overall development process, including installing and integrating portfolio management, platform management and a single-product stage-gate system, across its hardware divisions has been a five-year journey.

Has the journey been worthwhile? The company has begun to generate tremendous synergies between divisions. Not only did engineering become much more productive but so did all of the related functions of Procurement, Production Control and Materials Management. The reuse of just one subsystem across a company the size of IBM can have far-reaching effects. Across its computer product families, IBM has achieved a 50 percent reduction in part numbers. The cost of those that are shared between groups has been significantly reduced through better volume purchasing agreements. Different internal analyses have translated these improvements into a 3-percent reduction in base manufacturing costs for IBM’s hardware systems.

Additionally, the various commodity councils formed to operate between product families have identified numerous opportunities to employ industry standard components instead of IBM-made commodity parts. IBM has found that it is realizing a 15-percent cost improvement in buying these components from external vendors as opposed to making them internally. Perhaps most important, this has allowed IBM’s managers and engineers to more clearly focus on designing systems and building only those parts where they can clearly add value.

The impact on metrics usually associated with product development has been enormous. By the end of 1997, IBM was spending 42 percent less on new product development than in 1994 although its revenues were considerably higher. Figure 10 shows further results in terms of reductions in time to market and development expense.
Whereas IBM lost approximately $8 billion in 1993, in 1995 it had net earnings of approximately $4 billion, and in 1997 about $6 billion. Time to market has been reduced on average by an amazing 80 percent across all products, in all divisions. While many factors have come into play, executives within the corporation consider the company’s embrace of platform thinking and active management to be an important element of this success.

**R&D’s Role in Platform Development and Renewal**

R&D managers have a central role to play in guiding the development and renewal of the corporation’s product platforms. Principal areas of activity include:

- **Defining Platform Architecture.** The IBM example shows that while any particular generation of a product line may have a product architecture (comprising subsystems and interfaces), the greatest advantage is achieved if the corporation also views major subsystems and interfaces themselves as the product platforms that may be shared, and leveraged, across product lines. While this is traditionally seen as a way to share parts and reduce costs of goods, a focused R&D effort on making internally developed subsystems world-class will lead to improved functionality in a systematic manner across all product lines that employ those subsystems.

The R&D executive must ask engineering managers of each product line to articulate the product architecture for each respective area. The R&D executive must then bring these persons together to share their architectures, and from that discussion, to identify which subsystems are shared today (which may be none!) and which should be logically shared and leveraged across the corporation. These are nontrivial exercises. For those corporations that find virtually no commonality of subsystems between product lines, the R&D executive must set a starting target: perhaps 10 percent in the first year, 20 percent the next, and so forth, so that after five years, platform design and common subsystems become a standard way of thinking among new product engineers.

- **Defining Product Strategies that are Platform-based.** Many product development processes ask managers to define one product at a time. Platforms cannot be approached this way. Rather, platform management requires that R&D managers define several or more products that will share architecture, have common subsystems and interfaces, and by inference, be manufactured with a process that is in the main standardized. It is also a useful discipline to have teams forecast the additional engineering cost for each specific product relative to the development budget for the underlying common product and process technology.

The R&D executive can also challenge development teams to address one or more new market applications with each round of planning, for this leads to business growth. At a higher level, the R&D executive can challenge corporate staff to enhance development processes to embrace platform thinking. Such initiatives will force development engineers and product managers to reach beyond incremental innovation.

![Reduced Time to Market](image)

*Figure 10: IBM’s metrics for product development reveal substantial improvement between 1993 and 1997.*
Defining Platform-focused Organization: Perhaps the most sensitive of all issues, the R&D executive must assess the structure of the development organization and how it helps or impedes the sharing of common subsystems. Our own thinking in this area has changed substantially in the past year based on exposure to the thinking of highly innovative organizations. We see a number of R&D executives establishing subsystem development groups, where each group is responsible for current and next-generation development for its particular subsystems.

The focus of the platform groups is to achieve superior performance, scalability and cost advantage for their chartered subsystems. Their job may include basic research, but mostly it is to develop robust subsystems to serve the needs of product line engineers. Then, applied engineering teams within business/profit centers work with a different set of objectives and schedules to build specific products, performing the value-added development needed to supplement common subsystems. These engineers integrate the subsystems required for their product lines and add value to meet specific market needs as rapidly as possible to achieve time to market advantage. The platform subsystem groups are not the rebirth of central R&D. Rather, we view them as similar to the establishment of centers of excellence working toward very applied, tangible results. Each corporation must determine those structures and coordination processes best suited for its own culture.

Platform Principles for Your Own Company

The story told above is one of IBM’s use of platform management to drive unit costs down, increase gross margins while all the time reducing time to market. However, the story does not directly address creating new platforms for new markets. This, too, IBM is achieving today in its mainframe and services businesses, with considerable success. Platform thinking for physical products has played an important role in the company’s transformation of its traditional mainframe business to a Web server business. Moreover, management is currently in the midst of understanding and applying platform concepts to its e-services businesses.

Now, let us address new platform opportunities in your own company and how these opportunities can fuel business growth. Building the business case for the development of product or service platforms requires that you consider four key financial elements:

- Growth in revenues by addressing both new and current markets.
- Reduction in the “variable costs” required to create specific products derived from a common platform(s), both in terms of engineering costs, manufacturing ramp-up costs, and ongoing per-unit production costs.
- Additional “fixed costs” for the design of the common platform(s) and integration of various components and subsystems within that design, and/or the development of required interfaces to users and other systems.
- The further “fixed costs” for developing new competencies or licensing those competencies needed to build the feature, performance, price, and quality goals of the new common platform.

The bottom line is that, if pursued correctly, platforms can help the business grow and increase profitability. Platforms do not come free, however. As you try to make this argument within your own corporation, we expect that you will invariably run into several basic counter-arguments.

You may run into the argument that platforms are “too expensive” to engineer. While the initial development of product platforms may be more costly than the development of a single product, and may take longer, the benefits of faster and less costly derivative products based on strong platforms are compelling. We presented metrics for integrating engineering costs and product revenues in a Management Science article published several years ago.

As you consider platform costs, you must think about the returns on the stream of related products as opposed to single products. Unfortunately, many of the “hurdle rate” resource allocation processes in corporations focus entirely on single products. IBM no longer uses this as its only method of resource allocation and rewards. What matters most is the stream of products based on the current or newly developed platform architecture, and how these products collectively enable the corporation to achieve and sustain market share.
For example, let us take an example drawn from a company competing in the industrial equipment arena, doing about $200 million in sales annually with sales growth at approximately 10 percent, and a gross margin of about 50 percent. The engineering development cost of each new system was approximately $3 million, growing at 10 percent a year. A proposal was created to develop a new product platform—comprising an architecture, scalable computing hardware with real-time data acquisition capability, a real-time software development environment, and a set of common software applications suited for the firm’s target markets. All this would cost $10 million to develop. That platform would serve as the foundation for a stream of products for the next five years. These systems would have far greater capability than current-generation equipment.

Conservative estimates showed that each new product could be developed for about $1 million on average, and that four new products could be developed using the platform, as compared to just two new products without a common platform. Marketing experts expected that the new technology would produce sales growth in the range of 20 percent per year. Reuse of common parts and pieces would improve gross margins conservatively by 5 percent (to a level of 55 percent). Two new products would still be developed in the first year using the existing technology while in parallel with the development of the new platform. Products based on that new platform would emerge from the second year onward.

Figure 11 shows the projected P&L for doing business the old way relative to the new platform approach. However, executives had difficulty getting past that initial $10 million hurdle. It was only when one of the firm’s competitors announced that new products based on a similar approach would be forthcoming that this company’s management took decisive action.

It is essential that you bring your chief financial officer into the circle of platform understanding. First, as IBM’s recent history shows, the reuse of the common parts and pieces associated with strong platform architectures allows the corporation to realize the full benefit of volume procurement. IBM is not the only case. Black and Decker drove nearly all of its consumer power tool competitors out of business in the 1970s with an amazing 50 percent decline in cost of goods through common architecture and parts for motors, armatures, cords, and many other components. Hewlett-Packard enjoyed tremendous margins in the 1980s by having common interfaces across its ink jet printers. The same print cartridges worked across many different printer models and, being consumables, were produced at tremendous volumes with wonderful margins. Similarly, EMC used disk drives from the PC industry in its large-scale storage systems and was able to enjoy a 25-percent costs of goods advantage in million dollar systems over its competitors during the 1990s.

What these cases do not fully disclose is the force required to mandate reuse across each respective corporation. Development organizations tend to be smokestacks in design, working only within divisions and on single-product lines. This creates barriers to sharing of technologies and concepts between groups. The sheer amount of existing products and customers in large corporations creates an inertia that is hard to surmount. Like IBM, you may first have to inventory part numbers, manufacturing processes and computer systems just to understand how hard the problem is in your company, as well as the opportunities that exist for improvement. IBM used newly hired MBA’s to perform this work over three months. Other corporations have used consulting firms, even professors! To change development processes and organization cutting across business units, however, requires the commitment of your CEOs and CFOs.

Any product or service platform, no matter how robust, is invariably subject to obsolescence. IBM has learned this across its entire range of computer systems. Leading
financial services firms are now wrestling with the Web and its impact on their traditional telephone or branch office operations. In industrial and medical equipment, our research has shown that platforms seem to “run out of gas” every ten years. In more traditional software products, such as databases or programming tools, the durability of underlying architectures and designs now seems to be three-to-five years. In newer Web services businesses, firms are overhauling their business strategies, the structure of their content, and their content acquisition processes on a yearly if not quarterly basis. We believe that most companies can systematically and effectively renew their respective platforms, but the appropriate processes, organization and thinking need to be in place.

The prudent strategy is to make platform renewal—i.e., the development of new-generation architectures, subsystems and interfaces—a continuous process in R&D. This takes fortitude and persistence, together with the willingness to challenge conventional organization and processes. 

Acknowledgments

The authors wish to thank IBM leadership for sharing its data and experiences. In addition, we dedicate this article to our friend and colleague Al Bean who has been a motivating force behind our research and that of many others over the years.

References and Notes


5. For those readers interested in understanding platform management for systems manufacturers, see our description of EMC’s platform approach that appeared in Meyer, M. H. “Revitalize Your Product Lines Through Continuous Platform Renewal.” Research & Technology Management, March–April 1997, pp. 17–28. Since that publication, EMC has focused aggressively into value-added software and services, and only recently, announced a third-party software developer’s toolkit that allows other companies to easily build extensions to its own storage systems—a first for EMC.
6. The authors have developed a series of templates that we use in teaching and research to help teams develop platform strategies and implementation plans. These templates may be freely downloaded and used for planning your own development activities from the author’s Web site at Northeastern University. www.marchmeyer.com

Reprints

MEASURING AND IMPROVING THE PERFORMANCE AND RETURN ON R&D

Fifty-four RESEARCH • TECHNOLOGY MANAGEMENT articles on this subject are now available in paperback. To order, see inside back cover.

3000 Raw Ideas = 1 Commercial Success!
Assessing the Value of Your Technology
Metrics to Evaluate R. D. & E
Measuring R&D Performance—State of the Art
Evaluating R&D Performance Using the New Sales Ratio
Measuring R&D Effectiveness
The Strategic Measure of R&D
Visualizing, Measuring, and Managing Knowledge

Measuring R&D Productivity
Why Some R&D Organizations Are More Productive Than Others
Winning Businesses in Product Development: The Critical Success Factors
Reengineering for Revenue Growth
Managing R&D as a Strategic Option
Applying "Options Thinking" To R&D Valuation
. . . AND MORE

January–February 2001

39

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.