More to Do: Less Time to Do It

New product development.

Robert W. Hall

Because operating excellence is coming to be expected, poor cost, quality, or delivery will knock a company out of a market, but having it indicates only that a company is competitively fit. If several competitors in an industry are no longer widely differentiated in operating capabilities, in the mind of the customer the primary differentiator between them is their offerings. Attention frequently turns to New Product Development (NPD).

Likewise, it has long been known that many costs — and many process problems — are best solved at the design stage. Ford Motor Company is thought to be the source of the adage, “Seventy percent of product cost is determined by the original design.” The admonition is sound. Problems designed out don’t need to be solved later.

Interest in NPD is not confined to the older advanced economies. For instance, Lucky Goldstar, a Korean company, several years ago deployed an NPD process called Vic21, recently upgraded into Vic21+. In approach, training, and tools it resembles anything that one would expect from a world-class company. While Asians now have a financial crisis, more and more Asian companies see that competing on cheaper versions of old designs isn’t a way out of it. Industrial R&D is becoming global, and so are many companies’ NPD processes.

World-class NPD has become necessary to remain a world-class company — or to participate in a world-class supply chain. For the past ten years, *Target* has sporadically reported developments in NPD practice. They have come faster and faster, and so many have been advocated that a single organization cannot implement all of them. In most companies, NPD is still greatly affected by our experience with it a decade or two ago.

How NPD Evolved

Capturing the how-to details of innovative development is difficult. Even technology historians find it easier to describe individual feats than full innovative processes. For example, Thomas Edison’s personal exploits are well-chronicled, but how his laboratory worked is still being pieced together from its musty files.

A hundred years ago, formal NPD processes were limited to engineering-prototype-tooling-launch, a sequence that is still the legacy mind set of NPD for many of us in manufacturing and engineering. Market research was mostly a bet on intuition. A wide gulf remains between NPD as market development and NPD as product development. However, NPD potentially affects all parts of a business, or of a supply chain enterprise. New technology and manufacturing is only part of it.

After researchers began to nose into what we now call market research in the 1930s, the marketing side of NPD began to take shape. A classic was Paul Lazarsfeld’s “adoption studies” for hybrid seed corn among farmers, the forerunner of today’s product life cycle concepts. Other studies compared successful new product introductions with the flops, and found that “market pull” NPD was twice as likely to be successful compared with “technology push.” For a few decades thereafter, NPD in many large companies appeared, from the viewpoint of production and engineering, to be “dominated by marketing.”

In the 1960s and 1970s, market research became the first step in the over-the-wall NPD process sequence. Project management tools began to nudge NPD toward concurrent activity. Team-based “skunk works” projects were widely praised. Occasionally a hot innovation emerged from "bootleg projects" (budget stolen from authorized projects). But they weren't the norm, and it proved difficult for organizations to break out of the over-the-wall mind set.

Moore's Law forced the NPD process into compression — packing more into less development time — as companies raced to field hardware and software using the finest-scribed silicon available. Time-to-market — sometimes called time-to-money — became the buzz. Concurrent engineering teams began to displace over-the-wall organizations. NPD tools multiplied, and many became high tech. Using computer aided design, intelligent design, computer modeling, simulation testing, rapid prototyping, and other tools, much more can be done faster.

Much more must be done too. Technology is more complex, and customers demand that new products be free of significant flaws "right out of the box." Today if a company launched an innovation like the original, ungrounded car radio that could, and did, burn up entire vehicles, it would condemn itself to legal inferno.

NPD is a many-faceted subject, viewed in many different ways. Since it often involves both technology development and changes in business processes, it's highly integrative, cutting across many business functions, and sometimes across multiple companies of a supply chain. Consequently NPD is also a learning process that can easily drift. The management challenge is to keep NPD directed and integrated.

If successful as an innovation, NPD changes peoples' lives. For example, one of the criteria for judging winners of the Pace Award for innovation among auto suppliers is whether they change customers' processes and disrupt life for competitors. Most NPD outcomes aren't award-winning, many die in infancy, and some seem to be undeclared failures that drag along forever. In most companies, an important part of managing NPD is to focus effort where it will pay off, and stop projects if either the technology or the customer needs lead or lag the project. NPD management wisdom, but hard to apply, is to feed the winners and kill the losers.

NPD as a Business Change Process

Someone referring to their NPD usually means the procedures described in their NPD guides or manuals: Market research methods, financial hurdle tests, target cost methods, engineering tests and data required, checkpoints (or "gates"), certifications, methods of organization, scheduling and budgeting systems, and on and on. The manuals become thick because NPD is a process to develop significant changes in business and technical processes.

If it involves only an engineering change, a product launch isn't usually called NPD. For example, offering bolts in a different size package creates a new end item number, but changes in technical and business processes are trivial.

Likewise, companies that engineer-to-order, like Bodine Electric or Hoffman Engineering, may create a new product configuration with every order, but each order isn't an NPD. "Real NPD" fields a notably different "offering" to customers, and it might not be a specific piece of hardware, but a capability like the mass customization of product configurations. In a service company, like a bank, a different type of loan package is an NPD, but there it is obvious that technology only enables the real offering.

At the extreme, innovative NPD becomes the reinvention of a company, as with IBM and the personal computer, which changed life for both IBM and most of the rest of us. As a technical challenge, the PC was nothing special, but it thoroughly disrupted IBM's business processes — different types of customers, different logistics, and new non-IBM marketing channels.

The scope of changes in business processes suggest the time, cost, and risk of an NPD project. When NPD proposals (in cases) are examined in business schools, the technical risks are seldom thoroughly reviewed, but the business risks and rewards are. Changes that "don't fit the business" are not pursued. Translation: Radical reinvention of established business processes to serve a novel offering to different kinds of customers is risky, and may cannibalize the existing business. If the rewards are promising, an entrepreneurial start-up can build new business processes faster than morphing the old ones.

For example, Alto, the forerunner of the Apple, was developed by Xerox PARC in Palo Alto, CA, but Xerox didn't bring it to market. People chuckle at that decision now, but in retrospect, it was probably easier for Steve Jobs to invent new business processes and take the risk than for Xerox to do it. A different example of the same problem is Texas Instruments examining whether to machine artificial hearts, then abandoning the idea because the legal liabilities were in a range and field that TI and its insurers were not accustomed to. (Dow-

A Quality Function Deployment Table That Has Been Used for Training Purposes

Figure 1. *Originally contributed by Chris Fosse, now of Deltapoint.*

Corning and the breast implants were an instructive example.) A shallow-pocket shop would have more to gain with less to lose.

Timing and Integration

Assuming that the risk of a new offering is acceptable, the issue shifts to launching it well and at the right time. To compress a great deal of change into a short time, a multi-functional, and sometimes multi-national project team must operate on parallel paths.

The name "concurrent engineering" is a legacy from regarding NPD functionally, not as a change in business processes. If major business processes will be affected by a project, "concurrent reengineering" might be a better term. For example a current issue in HDTV development is how to get the cost reduction learning curve going — how both broadcast stations and consumers can initially finance the equipment — a situation reminiscent of Xerox's introduction of the copier, in which the breakthrough then was charging by the page.

Most managers have recognized that NPD should be integrated across all functions rather than a jumble of different, loosely coordinated processes, but integrating an NPD process is not easy, so practice lags behind concept. In general companies have found that virtual NPD teams work if they first have a face-to-face meeting so that key players get to know each other and agree on the mission of the project. After that, they can work on line at a distance. Subsequent face-to-face meetings occur at key decision points, or when confusion arises.

Tools for NPD process management increasingly assume the use of an integrated NPD process. Learning to use them well is more than just adopting a technique. A culture shift to work as an integrative team takes time.

Integrative Tools

Practices useful for improvement of NPD processes range from virtual teams to experimental design, so this review is necessarily limited. One of the oldest tools is scheduling and budgeting packages, now available in abundance, and one aspect on scheduling is becoming fundamental. Leading companies for several years have pegged NPD project schedules to pre-planned "gates." A gate is a major go-no-go review point. A typical NPD schedule template is anchored by three to seven of them. Typically, the first gate is approval of a concept for development; the last one, approval of commercial launch.

Lack of discipline to "hit the gates" is a common problem. One reason is excessive "mission creep" — shifting or uncoordinated objectives. A second is that team members are not clued in to have appropriate data ready for a rational decision at gate time. When project personnel understand what they must provide to make a good decision at each gate, they are less likely to dally.

Another tool that has grown in use in North America for about 15 years is Quality Function Deployment (QFD), sometimes called the House of Quality. Figure 1 shows a QFD table from a *Target* article on the subject several years ago.[1] The core of QFD is the relationship table that "forces" a team to integrate the needs of the customer with the process design that will satisfy those needs. Although detailed, the table is easy to comprehend, so a team can generally walk through a superficial QFD exercise without difficulty.

However, by itself, a QFD table does not ferret out the subtle needs of customers, nor reveal ingenious possibilities for process design. It's only a framework to integrate product and process design with customer needs as the QFD team understands them. Note that the table in Figure 1 concentrates on design of a golf club. Not included are the processes for marketing, distributing, or after-sales service of the golf club.

While QFD is a format to integrate a mass of detail to better serve the customer, to generate the detail, teams are on their own. Difficulties of detail integration are illustrated by the issues arising from the use of various "design for" methodologies.

DFX: Integrative Design

DFX is a cutesy acronym meaning "design for everything." Some "design for" methodologies with established design rules are: Design for Assembly. Design for Manufacturability. Design for Test (mostly electronics). Design for Mass Customization. Design for Maintenance. Design for the Environment. Design for the user is still important too.

Roll the first three of these "design fors" together with a little target costing, and you can call the result Design for Cost. Stir in design for maintenance and for the environment and you can call it Design for Lifetime Cost. A few acronyms into this, a design engineer will protest that DFX is what engineers have always done.

But engineers haven't always done DFX. They have designed the best they can within the limitations of their

Design for Assembly/Manufacturing

Modularity:	Interchangeable sub-assemblies Reuse design platforms when possible Minimize number of levels of assembly Minimize assembly space required Minimize material handling equipment requirements
Minimize part counts:	Fasteners, cables, harnesses, etc. Reuse of parts or materials Software packages
Access:	Minimize number of re-orientations of parts and assemblies Visible, reachable insertion points Use existing or common tool sizes Allow for tool and machine access in fabrication Allow for hold points for fabrication
Fabrication:	Design curved surfaces to tooling and CAM software capability Tolerance to process capabilities Avoid extraordinary requirements (Examples: Long depth/diameter holes; complex geometries) Avoid difficult joining of dissimilar materials.

This is a brief summary of key points from several lists. It's mechanically oriented, but may serve as a "starter" check list.

Figure 2.

Life Cycle Design Considerations

Process Stage	Environmental Items to Consider
Raw Material Acquisition	Habitat alteration Accidents Resource renewability
Material Manufacturing and Fabrication	Energy usage Net water consumption Pre-consumer waste recycle % Airborne emissions Waterborne effluents Solid waste generation rate
Packaging and Distribution	Product/package reuse Similarity of product/packaging materials Reuse of containers
Recycle/Waste Management of Product in Field	Remanufacturability Disassembly potential Recyclability potential Waste to energy ratio Material persistence (biodegradability) Toxic material mobility (landfills and incinerators) Toxic content Inhalation toxicity from combustion and incinerators Incinerator ash residue Landfill leachate (aquatic) toxicity

Figure 3. *Adapted from a table used by Pacific Northwest Laboratories and Battelle Institute. This is a summary table because a complete one is impossible. A few major points are summarized from an infinity of possible considerations. A software package called LCAdvantage is used to help guide people through an analysis. LCA is Life Cycle* Analysis.

tools and experience. Unfortunately, no one lives long enough to acquire experience on everything important, so tools and techniques must extend individual competence.

The most common integrative techniques are NPD process checklists and design reviews. Design for Assembly has become well known. A short list of some common issues in Design for Manufacturing is shown in Figure 2. An example of a checklist for Design for Environment is shown in Figure 3. Such checklists are only reminders, of value only if designers have time to learn how to follow up. That is, designers have to construct procedures applicable to themselves and their projects that enable them to find and use information at the right time. A beginning database for Design-for-Manufacturability might consist of little more than available equipment and tooling. Like many other things that sound simple, it generally has to start small and build up.

On the other hand, a design review opens a design to critique by people from different perspectives, and who bring knowledge with them. For example, a design-for-manufacturability review lets manufacturing engineers and shop workers tear into a design to make it more producible, as Lantech, a company in Louisville, KY, did with its first shop "kaizen" on a new product design six years ago. It has continued to build this experience into design "rules" since, but continues the "kaizens."[2]

A series of design reviews by people of different viewpoints takes time and the recommendations at each stage are apt to conflict. However, human reviewers have insights that go beyond tools and checklists, and solving any potential problem earlier is better than later. In addition, reviews help breach the walls between groups. When designs are not handed down from on high, they are more likely to be accepted by everyone who must work with them.

Building Libraries

Some kinds of design issues, like a performance trade-off between acceleration and fuel economy, are obvious and a part of a project's design objectives. It's the "little details" outside the personal knowledge of designers that cause unforeseen problems. Finding and fixing those is the objective of DFX. Beyond the checklists and reviews, databases or libraries of today's software tools incorporate a legacy of design knowledge.

The advantages of CAD, CAM, simulations, models, and other packages are taken for granted: software is easy to store and transport, which allows simple design changes to be made quickly. So many more design options can be generated and evaluated that a common complaint is that designers want to play with the tools instead of getting on with it. But when we really want to, the tools also help to develop a design quickly.

A far-out ideal is to download a software-tested design to production without testing prototypes. If designs are so well characterized that no physical testing is necessary, as with mass customized variations, that is possible, but prototype testing is usually necessary because every condition in actual use isn't built into the models. More realistic, but a stretch, is "paperless" design of a prototype. Little or no paper is required to convert the software to a hardware prototype, with model testing so thorough that few revisions are necessary after testing and human review of the design in hardware.

Many companies now realize that their design libraries are one of their major assets. For example, in the automotive air bag business, being able to pick a likely algorithm from big repertoire of crash test profiles gives one a leg up designing a system for a new vehicle.

As the library of these tools builds, some of the DFX knowledge is incorporated in it. A few advanced systems even have intelligent agents (like web crawlers) gather relevant information from a database network about design parameters, customer data, test histories, tool and machine capabilities, software capabilities. etc. Neat idea if you have good data in the bases, and if the agent's hits are relevant rather than annoying.

Managing a more complex design process presents both old and new challenges. An old challenge in a new context is engineering change control — old fashioned configuration management. While it's marvelous to quickly try all kinds of design alternatives, the overall effort must be systematically coordinated or several designers of both hardware and software may be making contradictory changes at once. To check whether changes integrate, they must be regularly tested in combination in an integrated package.

Digital design has many advantages, such as running a color coded stress analyses during design, or designing and executing microprocesses that were not possible before computers. It also has new hazards, one being that human perception of a design on a screen is not the same as on paper, and neither are the same as sensing a physical form that thuds when it is dropped, and if kicked, causes pain in the toe. The mental approach to digital design is different too. One can easily assume that the models and packages being used for design are always complete, so designers optimize on a limited set of design criteria while overlooking critical "design fors." There's still something to be said for plodding through a manual design, thinking as you go, analogous to a person on foot noticing details about the terrain that are oblivious to a driver zipping along in a car. However, the advantages of digital design are so great that we're not going to return to slide rules.

The Model T Wasn't Designed This Way

The ability to quickly try numerous design alternatives is evolving into a new kind of design. While far from commonplace, there are programs in infancy that operate on a complex set of data and conflicting criteria, then let an optimal design emerge. The basic idea is to break a design into modules and set up a program to try alternatives for each module: sizes, configurations, materials used, and so on. Then try "a gazillion" different combinations of these in an integrated package with the whole run through test programs. The process searches for the design combination that will best fit a set of pre-determined overall design criteria.

This is a Darwinian process. Bad design combinations are killed off. Good ones are kept. The best features of the survivors are mated and mixed, and sometimes the bad features are mutated and tried again. From the offspring of this process, losers are again culled out until a "great" design emerges. No one can say that it is the best one possible, but it is better than might be expected from a slower design generation process.

The Bios Group at Santa Fe, NM is a pioneer in emergent design. One of their applications is with Honda's North American design styling group, so that each design generation is evaluated on aesthetic appeal (but not on DFX — not yet). The process picks winners, kills losers, and mutates ugly features, while updating a genealogical history of design features based on aesthetics. The objective is to speed the evolution of design based on subjective criteria by increasing options viewed while decreasing artistic debate time.

The performance elements of aircraft have always required trade-offs in design: fuel economy versus speed versus payload versus lift versus drag, and so on. Bios has also created a "multi-objective optimization" program to tackle that one for Boeing Commercial Aircraft. An airplane is too complex for a single software package to generate a meaningful overall design by itself, so this program guides different departments through a rapid development of concept designs by integrating design modules and testing each combination using everything from finite analysis to Navier-Stokes air flow testing. As before, the good are kept, the bad ones culled, and some non-performers are mutated for retrial. No DFX is incorporated, but a concept design is developed much faster.

Speeding the Learning

NPD is an organized learning experience — problem solving by gaining feedback from a broader environment — from customers, from suppliers, and elsewhere — and making adjustments as quickly as possible. No matter where an organization stands in developing its NPD process, increasing its learning scope and speed are overall performance criteria for improving time-to-market.

1. Dave Hendrickson and the *Target* Staff, "Product Design as a Team Sport," *Target*, Vol. 6, No. 1, Spring 1990, p. 6.

2. Pat Lancaster and Ron Hicks, "Using Kaizen to Improve Designs and to Speed Development — How Lantech Kaizened a Problem Product," *Target*, Vol. 11, No. 5, Sept./Oct. 1995, pp. 24-29.

Robert W. Hall is editor-in-chief of Target *and a founding member of AME.*
