Harvard Business Review

OnPoint

ARTICLE

IDEAS AT WORK

Getting the Most out of Your Product Development Process

by Paul S. Adler, Avi Mandelbaum, Viên Nguyen, and Elizabeth Schwerer

Included with this full-text *Harvard Business Review* article:

Getting the Most out of Your Product Development Process

The Idea in Brief

Despite meticulous planning, are your company's product development projects stuck in seemingly permanent logjams, running months behind schedule? If so, you may be viewing product development as a list of individual projects. But product development is a complex *process* that can be streamlined and accelerated.

To get your new offerings to market more quickly, you need to know how many projects your company can handle—which means attending to employees' and departments' capacities and workloads. And that requires a strategic view of your *entire* product development process—not just individual projects'.

By replacing project management with **process management**, you exploit similarities across project tasks through standardization and continuous improvement—without destroying creativity. You also relieve bottlenecks, finish projects faster, and smooth out workloads. Results? A 30%–50% reduction in time to market.

The Idea in Practice

To apply process management to your firm's product development:

Draw a processing network model showing all departments involved in product development, all tasks they perform, and information (blueprints, test results, verbal authorizations) flowing among departments.

Notice how most projects reside simultaneously in several departments in various stages of completion (e.g., engineering and technical services are working on a prototype, while marketing completes its plan). The model enables you to see the "forest" (the high-level view of your product-development process) rather than just the "trees" (individual projects).

Circulate a process questionnaire to all product developers, asking about **projects** ("How many types of projects does your group handle?"), **resources** ("How many hours do people in your group work every week?"), and **processes** ("How many iterations does each task require in a project of average complexity?"). Analyze your company's development experience over the past few years. Summarize results in a Resources and Requirements table and determine each department's capacity constraints.

Create a utilization profile comparing each department's available hours per year with hours required by project tasks and other activities, such as administration. Look for surprises—a large share of the workweek consumed by non-project work, groups above full utilization, groups with widely varying workloads. You'll begin to see why many projects seem to take forever.

Build a computer simulation model predicting how long projects will take to complete. Using data from the preceding steps, quantify variations across types of projects in their sequence of tasks, number of task iterations, and rate of new project starts. Analyze the resulting average-completion-time graphs. For example, perhaps 10% of new products (regardless of their complexity) take 140+ weeks to complete. Watch for projects with *low* market potential that are consuming the lion's share of management time and energy.

Improve balance between resources and workload, for example, by:

- adding resources to bottlenecked departments

- automating bottlenecked steps

- limiting the number of projects under way simultaneously

- starting new projects only when resources are available

- reducing the number of urgent projects that interrupt work

- documenting best practices to reduce new projects' setup time and decrease variation in time to perform similar tasks.

Calculate each idea's costs and benefits, identifying investments that will generate the biggest payoffs.

Implement changes, reporting results on a trial basis and fine-tuning as needed.

▶ Example:
ConnectCo, an electrical-connectors producer, trimmed its project portfolio from 32 to 22 ongoing projects—completing 30% more projects than its annual average. It also accepted only eight *new* projects that year—60% less than usual. Results? Average development cycle time decreased 35%, and team members pinpointed hidden bottlenecks, skill shortages, and best-practice template inadequacies.

The lessons of lean manufacturing can help companies develop new products faster.

Getting the Most out of Your Product Development Process

by Paul S. Adler, Avi Mandelbaum, Viên Nguyen, and Elizabeth Schwerer

Process management has revolutionized manufacturing. Companies around the world have reduced cycle times in their factories by studying each step in the manufacturing process and fluctuations in workloads for ways to reduce variation and eliminate bottlenecks. The product development process can be streamlined in much the same way.

Indeed, we argue that general managers who need to know how many projects their development organizations can handle—and how quickly those projects can deliver new products to market—must think in terms of managing a process. Most managers, however, think of product development simply as a list of projects rather than as a complex operation with a given capacity and workload.

The initial reaction of many managers to the suggestion that product development could benefit from a process management approach is, "Product development is not manufacturing. It is mainly knowledge work. The tasks are not nearly as repeatable as they are in manufacturing, and standardizing the work would

kill creativity." Yes and no. Each development project involves unique challenges that require unique solutions. But there is a lot of work in product development and in many other kinds of knowledge work that is not unique. Many tasks and sequences of tasks are the same across projects. Process management exploits those similarities through standardization and continuous improvement—without destroying creativity.

During the past eight years, we have studied a dozen companies that have started to apply process management to product development, including Raychem, Motorola, Harley-Davidson, Hewlett-Packard, General Electric, AT&T, Ford, General Motors, and NEC. These pioneers have made three discoveries. First, projects get done faster if the organization takes on fewer at a time. Second, investments to relieve bottlenecks yield disproportionately large time-to-market benefits. Third, eliminating unnecessary variation in workloads and work processes eliminates distractions and delays, thereby freeing up the organization to

focus on the creative parts of the task. The result: Business units that embraced this approach reduced their average development times by 30% to 50%.

Process management is a particularly effective way to reduce the congestion that plagues organizations that undertake many projects at once and share staff and equipment across those projects. The typical project-management approach to product development, however, obscures the overall process. Consider the experience of a major computer-equipment manufacturer that we studied. To minimize the number of iterations, or rework cycles, in development projects, management had created cross-functional concurrent-engineering teams to identify and solve problems rapidly and early. But the development organization tackled so many projects at the same time that key people from engineering, marketing, and manufacturing found themselves working on five or even ten projects at once. To make matters worse, project managers tried to force their own projects ahead by commandeering resources, which delayed other projects even more. As a result, critical people in the development organization were unable to juggle the many demands despite 60-hour workweeks, and most projects ran late.

To avert such logjams, a large manufacturer of electronic components went beyond creating cross-functional teams and drew up an aggregate plan for all development projects. The plan ranked proposed projects by their strategic importance, taking into account the nature of each project (breakthrough, platform, derivative) and a rough estimate of the resources each would require. The company used this analysis to reduce and focus its portfolio of projects. Nonetheless, most projects continued to run months behind schedule, and the plan did not help managers understand why: At each stage of development, engineers had to wait for support technicians to run critical tests of prototypes. Although there were enough technicians to support the average workload, the actual workloads were uneven, and, as a result, the technicians often had long backlogs.

An aggregate project plan is a valuable tool for winnowing out margin-al projects and focusing a company's development effort on strategic priorities. Such a plan can also help ensure that the organization does not take on more projects than it can complete—a surpris-

ingly common problem. (See Steven C. Wheelwright and Kim B. Clark, "Creating Project Plans to Focus Product Development," HBR March–April 1992.) But project plans are only a first step toward faster development. To take the next, much bigger step, managers need to think of product development as a production process in which projects move through the knowledge-work equivalent of a job shop. This process view helps managers identify and solve congestion problems caused by mismatches between the workload of each subunit in the development organization and its capacity to handle that workload.

A process view can also help managers eliminate excessive variability in workloads, another cause of congestion. Variable workloads usually arise because an organization takes on new projects whenever good market or technical opportunities present themselves. As a result, in some months many projects start, and in others none do—a pattern that can create bottlenecks at crucial points in the development process. We have seen instances in which managers thought they were being prudent when the number of projects that they had assigned to the development organization required it to operate at about 90% of its capacity. If they had looked more closely at the variation in the total workload, however, they would have found that behind this annual average lay week-to-week fluctuations ranging from 80% to 150%. If those managers had reduced their planned average utilization rate to 80%, they could have reduced development times by 30% or more.

Finally, a process management approach can help reduce variability in the way specific jobs are executed. The benefits from eliminating rework cycles and abnormally long steps are often disproportionately large because those sources of variability delay not only the project in question but all projects under way.

Some development organizations try to avoid congestion by relying on autonomous, or dedicated, project teams, each of which works on one project at a time and has all the resources it requires. Such teams are common in software development, for example. But this approach is expensive because it means duplicating rather than sharing resources. In addition, congestion can still arise within such projects, especially if the project-staffing plan underestimates the amount of rework that the

Paul S. Adler is an associate professor of management and organization at the University of Southern California's School of Business Administration in Los Angeles. Avi Mandelbaum is a professor on the faculty of industrial engineering and management at the Technion in Haifa, Israel. Viên Nguyen is the Robert N. Noyce Assistant Professor of Management Science at the Massachusetts Institute of Technology's Sloan School of Management in Cambridge. Elizabeth Schwerer is a Ph.D. candidate at Stanford University in Palo Alto, California, and an instructor at Columbia University's Graduate School of Management in New York City.

team ends up having to perform.

The ConnectCo Case

To illustrate the steps a company can take to put a process management approach into action, we have created a case study of a fictitious company we call ConnectCo, a composite of several companies we have studied. For competitive reasons, those organizations requested that we not release the details of their product development processes.

ConnectCo, a producer of electrical connectors and adapters for industrial use, was under pressure to accelerate its development cycle after losing several potential contracts to a Japanese competitor with much faster product development. The principal charge of Con-

nectCo's product development group was creating new products, but it also undertook smaller product-line extensions and supported products already on the market. ConnectCo's development projects were not very complex. They usually involved one development engineer, one technician, and the support of several other groups. However, the company undertook many projects, and customers often demanded changes in performance standards.

ConnectCo had revamped its development process several years earlier. Management had established a formal product-development procedure specifying the activities necessary at each phase of development. The development organization had instituted cross-functional teams and implemented a planning process to

Congestion in Operations and Product Development

To: Steve G., Bernice W., Mike J., and Bill S.
From: Mark E.
Re: Faster development time

In our manufacturing plant, the lead time of a job is the sum of two components: the amount of processing time that the job requires and the amount of time it spends waiting for machines to become available. The time spent waiting at each machine increases with three factors: the planned utilization of the machine, the variability of the workload assigned to that machine, and the variability of the machine's processing capability. The graph and the equation show how these factors interact.
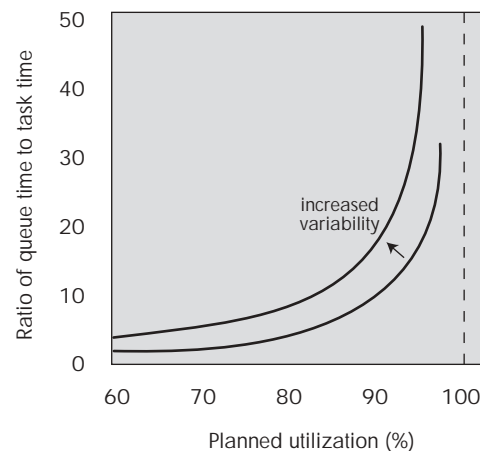
$$\frac{\text{queue time}}{\text{task time}} \;=\; \frac{1}{2} \;\times\; \left(\begin{array}{c}\text{workload} \\ \text{variability}\end{array} + \begin{array}{c}\text{process} \\ \text{variability}\end{array} \right) \;\times\; \frac{\text{planned utilization}}{1 - \text{planned utilization}}$$

If our products have to move through several backlogged workstations and if some tasks need rework, little wonder that our plant, which was operating at 90% utilization with high workload and high process variability, often needed to quote lead times nearly 20 times the actual processing requirements.

In product development, work centers are people rather than machines; workload variability is the variability in the number and type of projects taken on; and process variability is the variability in the amount of time and the number of iterations needed to complete tasks. If the number of projects we start implies a planned workload of 90% to 95% of capacity (which it usually does, even when we want to leave a cushion) and if the organization experiences both workload and process variability (which it certainly does), then it is hardly surprising that our project completion times are more than five times the critical-path prediction.

achieve a balance between the types and numbers of projects it undertook and the available staff.

Despite those measures, Mark Epstein, ConnectCo's general manager, still felt that he did not really know how many projects his development organization should undertake. The formal development procedure helped him predict the amount of work that each project would require. On that basis, the organization did not seem to be taking on too many projects. But Epstein lacked a tool for predicting when those projects would be completed. No matter how much extra time he allowed for unforeseen contingencies, more than half of the projects scheduled for completion each year remained unfinished. Some projects spent years in limbo. Recently, the development department had started to use project-planning software, and Epstein had been dismayed to discover that the average development time was more than five times the critical-path time: the minimum time—not accounting for delays or rework—that the company estimated a project required.

Why did ConnectCo's projects take so long? Epstein asked his development manager, Steve Gilles, to make a list of recent projects and categorize them by difficulty and duration. Not entirely to their surprise, Gilles and Epstein found that technical difficulty was not a good predictor of time to market. One product-extension project, the adapter AD325, had required only two person-months of work and yet had taken more than two years to get to market. A much larger, more innovative project, the AD3500, had been completed in less than a year.
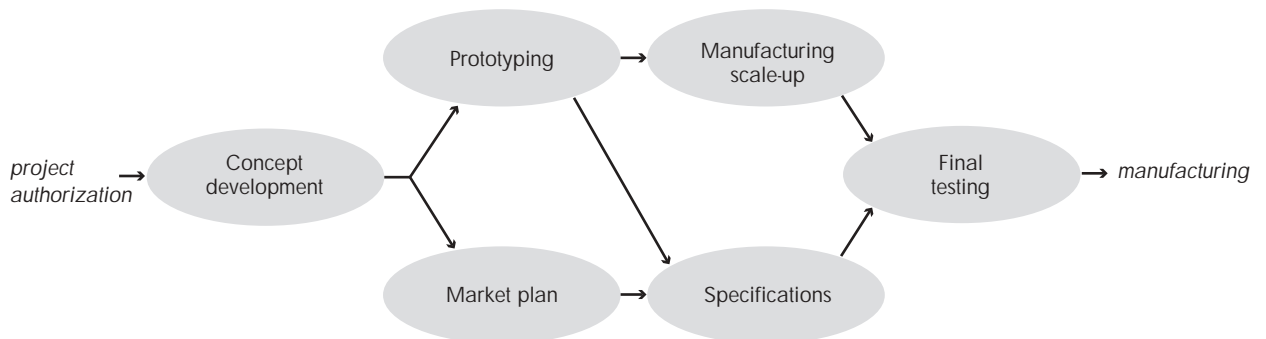
The AD3500 team had been led by a young engineer, Laura Murphy, who had proved herself to be an energetic and creative leader. To push her project ahead of the others, however, Murphy had needed very sharp elbows, and there had been complaints that the concentration of ConnectCo's resources on the AD3500 had slowed down other projects. Epstein concluded that the complaints reflected a real problem that ConnectCo had encountered many times before—and not only in development.

Developing a Processing Network Model

Epstein remembered seeing similar lead-time problems in his manufacturing organization. A consultant had helped ConnectCo develop a process-simulation model of the flow of products through the plant floor. The model, which took into account variability in orders and in processing times, showed that products usually spent the bulk of their time in a queue for equipment rather than being processed. It demonstrated to ConnectCo's managers why planning high levels of equipment utilization led to congestion and how expediting urgent jobs added variability and thus delays to an already stressed system.

Epstein sent a memorandum to his management team explaining how similar problems were causing the delays in product development. (See the exhibit "Congestion in Operations and Product Development.") Epstein and Gilles then set up a cross-functional process-improvement task force to build a model of the development process like the one created for manufacturing. To send the message that management considered the effort vitally important, Epstein and Gilles selected Murphy to

The Project Flowchart

project authorization → Concept development → Prototyping → Manufacturing scale-up → Final testing → *manufacturing*

Concept development → Market plan → Specifications → Final testing

Specifications → Final testing

head the task force.

The task force began by developing a conventional project flowchart that showed the six major tasks in the company's formal development procedure. (See "The Project Flowchart.") The group quickly realized that because the chart didn't identify the organization's resources, it would not reveal which were overutilized. Using the manufacturing model as a template, the task force came up with a new representation. (See "The Processing Network Model.") The network model shows that five departments contribute to the product development effort: engineering, marketing, technical services, specifications, and manufacturing engineering. Each department is responsible for several activities. For example, engineers are responsible for concept development, prototypes, final testing, and support and administrative activities. The lines connecting the departments show how test results, specifications, blueprints, verbal authorizations, and other information flow between them.

Unlike the project flowchart, the processing network model reminds managers that projects usually reside simultaneously in several departments in various phases of completion. For example, engineering and technical services may be working on a prototype for a project while marketing is completing its plan. Furthermore, each department in the organiza-

tion usually has more than one task queued in its in-box. On a given day, a technician may find requests to perform qualification testing and manufacturing scale-up for three or more projects. Such a model can also help managers see the numerous iterations that can occur in a project.

Instead of showing only the trees (the individual projects), the network model reveals the forest (the structure of the process). Epstein challenged the task force to build a quantitative simulation model that would help the company identify and assess various improvement options. New personal-computer software packages, he pointed out, have made it relatively easy to create such simulation models.

The members of the task force set about collecting the requisite data using a questionnaire for all the participants in the development process. (See "The Process Questionnaire.") The participants were able to answer some of the questions easily, such as how many people were in their particular group. Answering other questions, however, required a new perspective. These questions included "How many different types of projects does your group handle?" and "Within each project type, how many iterations are required to perform each task in a project of average complexity?" In tracking ConnectCo's development activity, the company's management-control system had focused on individual projects and people, not on

The Processing Network Model
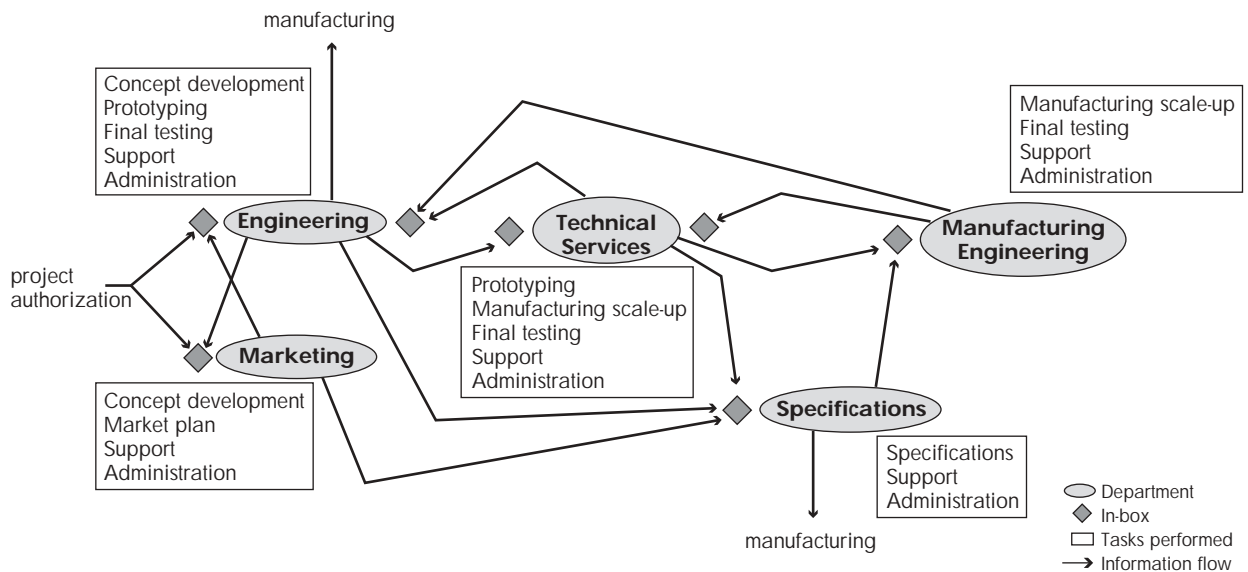
the kind of process-oriented information required in the questionnaire. To help the participants complete the questionnaire, the task force organized a series of workshops with people from each group to analyze ConnectCo's development experience over the preceding three or four years. The task force summarized the results in the "Resources and Requirements" table.

The task force now had the raw material it needed to determine the capacity constraints of each department. To this end, the task force estimated the planned utilization of each group by comparing the group's available hours per year with the hours required by project tasks (the product of the average person-hour requirements per project and the number of projects per year) and by other activities such as administration and support. The results of this capacity analysis are shown in the "Utilization Profile," which compares the rate at which the organization can develop products with the rate at which projects start.

The members of the task force were surprised by what they found. They had not been aware of the share of the average workweek that nonproject work was consuming. More important, they learned that several groups were near or above full utilization. Engineers, for example, were scheduled for an average utilization of more than 104%. When variations in project demands and in workload were added

to the picture, it became obvious why some projects took forever.

The utilization profile raised other interesting questions. For example, although no one doubted that the technicians were overloaded, the data showed that they had free time. Follow-up discussions with the engineers and the technicians revealed that the technicians were using this time to help out on several engineering tasks, including testing. Managers had not paid much attention to these informal but highly effective practices. The task force members concluded that this sharing of tasks was preventing even longer delays in product development.

Analysis and Options

This first phase of analysis indicated that with a little overtime and some sharing of tasks, the development organization should be able to complete the existing number and mix of projects. But the analysis did not predict how long it would take to complete those projects. To estimate the cycle time for a multistep process like development—especially one that involved as many iterations as ConnectCo's—the task force needed to build the simulation model that Epstein had proposed.

Adapting the approach that the company had taken in the manufacturing study, the task force used the data from the questionnaire to quantify the variation across projects

The Process Questionnaire

Projects

☐ How many types of projects does your group handle?
☐ How many new projects of each type does your group undertake each year?
☐ What tasks are involved in each project type, and is there a specific order in which they must be carried out?

Resources

☐ To which phases of product development does your group contribute?
☐ How many people are in your group?
☐ How many hours do they work in a week?
☐ What project-related tasks does your group perform?
☐ What nonproject tasks (administrative and support) does your group perform?
☐ How many hours does your group spend on each task?

Processes

☐ For projects of average complexity within each type of project, how many iterations does each task require?
☐ What is the probability distribution of task processing times and of number of iterations across projects?
☐ What proportion of projects in each type are easy, intermediate, and complex?
☐ How does each person decide which project or task to work on next?

in the sequence of tasks, the number of iterations, and the rate of new project starts. Soon the task force had a model simulating the flow of projects through the organization. With some tweaking, the team calibrated the model so that it reflected the general consensus on the distribution of completion times for the two main types of projects. (See the exhibit "ConnectCo's Historical Project-Completion Times.")

These graphs highlighted a point that had emerged in the data-collection workshops: While ConnectCo needed to improve its average development time, it also had to do something about inordinately protracted projects. In fully 10% of new product projects, the completion time was more than 140 weeks; and even though extension projects generally required only 365 person-hours, 10% of them took more than 100 weeks. Tracking those projects was a management drain—and an unjustifiable one because they were not particularly difficult or high in market potential.

In a brainstorming session, the task force generated an array of possibilities for reducing development time, which the group then assessed using the simulation model. First, there were many ways in which ConnectCo could reduce the average utilization of the departments where there were bottlenecks. It could add resources to those departments. It could reduce the average number of projects under

way at any time. It could train people in less burdened departments to perform tasks of overburdened departments. It could eliminate unnecessary steps. It could automate steps that had become bottlenecks. And it could reduce mental and physical setup times by improving the content and availability of project documentation.

Second, the organization could reduce the variation in the times required to perform tasks by creating best-practice templates. Expanding the development-procedure manual to include such templates would stimulate the sharing of best practices throughout the organization and would help bring newcomers up to speed more quickly.

Third, the task force considered ways to reduce the variation in the overall workload. Managers could set a limit on the number of projects allowed in the system at any one time. Perhaps development could operate a pull system modeled after the highly effective just-in-time approach used in the manufacturing plant. Under such a system, a new project could be started only when another was completed.

Finally, the company could rethink how it handled urgent projects. Expedited projects interrupted work in progress, resulted in extra setups, and increased variability in the process. Basically, there were two possible solutions: reducing the number of expedited projects or increasing the development organization's ca-

Resources and Requirements

	Engineering	Technical Services	Marketing	Specifications	Manufacturing Engineering
Full-time equivalents (FTEs)	5	8	3	3	4
Support (hours/week/FTE)	12	8	12	12	12
Administration (hours/week/FTE)	4	2	12	8	18
New Products*					
Average number of projects initiated per year: 10					
Concept development	12 x 4		6 x 4		
Prototyping	20 x 12	20 x 21			
Market plan			180		
Manufacturing scale-up		13 x 18			15 x 6
Specifications				200	
Final testing	30 x 9	25 x 12			10 x 6
Product Extensions*					
Average number of projects initiated per year: 4					
Prototyping	12 x 4	16 x 5			
Manufacturing scale-up		10 x 4			15 x 3
Specifications				8	
Final testing	20 x 3	16 x 3			18 x 2

Resource requirements for the average project expressed as hours per task times average number of iterations per task.

pacity.

The task force developed some rough cost-benefit calculations for each of its many ideas. The group quantified the benefits of faster development in terms of greater market share and longer product life. The costs associated with each scenario included the direct costs of resources and, in some scenarios, the revenues that the company would forgo over the short term.

Given existing budget pressures, the task force didn't think management would support a proposal to add expensive equipment or people such as engineers—even though the simulation-based calculations suggested that those investments would generate high returns. Instead, the group proposed two relatively modest investments that could have big payoffs. The first was to train technicians so that they could conduct more of the testing performed by engineers, who often took hours to program complex testing procedures. Technicians were already helping out, but with training they could handle most of this programming. The necessary courses were offered both within the company and at the local college.

The second recommendation was to limit the number of new projects under way at any time to 12: nine new products and three extensions. Currently, the company was starting about 14 projects—ten new products and four extensions—a year, but because each took so long to complete, there were often more than 30 projects in the system at once. The simulations showed that if ConnectCo instituted a pull system that allowed only 12 projects to be under way simultaneously, project starts would

probably fall by 10% to 20%, but each project would be completed much faster.

Murphy's task force found that together those two actions would cut average development times for both new and extension products by nearly 40%. Moreover, the time required to complete the worst 10% (the most protracted) of both new-product and product-extension projects would fall considerably. (See the exhibit "Estimated Improvements in Completion Times.")

Like many other companies, ConnectCo had tracked the hours that had been spent on each project each week by each person. But Murphy's task force concluded that those data did not help the company monitor and improve the development process. The group proposed that ConnectCo maintain a battery of new process-oriented measures. Those measures included load (the number of projects in progress each month); resource availability (the development resources available each month, net of administrative and support time); utilization (the monthly utilization level of each department); contribution (the time contributed by each department to each task during the month); process yield (the number of iterations required to complete the task successfully); and process efficiency (the ratio of actual time spent on the task to the minimum possible time as estimated by a critical-path model and best-practice templates).

The efficiency and yield data could be collected for each project on a monthly basis and then aggregated to characterize the degree to which each task was under control. The task force recommended that managers track not

Utilization Profile

Hours per Year	Engineering	Technical Services	Marketing	Specifications	Manufacturing Engineering
Available	10,000	16,000	6,000	6,000	8,000
Support	3,000	3,200	1,800	1,800	2,400
Administration	1,000	500	1,800	1,200	3,600
New products	5,980	9,540	2,040	2,000	1,500
Product extensions	432	672	0	32	324

Planned Utilization (%)
(100 x hours ÷ available hours)

	Engineering	Technical Services	Marketing	Specifications	Manufacturing Engineering
Support	30.0	20.0	30.0	30.0	30.0
Administration	10.0	3.1	30.0	20.0	45.0
New products	59.8	59.6	34.0	33.3	18.8
Product extensions	4.3	4.2	0.0	0.5	4.1
Total	104.1	86.9	94.0	83.8	97.9

only averages but also the worst decile in order to keep tabs on the projects in limbo. (See "The Process Reporting Form.")

Finally, Murphy's task force proposed that the development-procedure manual be augmented with best-practice templates for each task. Although some variance in a development process is inevitable because of each project's idiosyncrasies, everyone on the task force had been shocked to discover the degree to which the number of iterations and the time required to carry out a given task varied from project to project.

Decisions Taken

Epstein and his managers were impressed by the cross-functional-training proposal and quickly gave the go-ahead. But they had a harder time accepting the recommended move to a pull system. Going from more than 30 ongoing projects to 12 seemed risky. Although the simulation ultimately convinced them, Epstein was nervous about the transition. He decided to trim the number of ongoing projects to 20 over the next year and then reassess the number for the following year.

To that end, Epstein instituted a more rigorous review process for project proposals and asked his managers to review all projects that were near completion but had stalled. Epstein suspected that although those projects did not require much additional work, many of them had been caught in a vicious circle: Once a project acquired a reputation for being a problem, it was continually pushed aside by newer projects, especially by those whose leaders had sharp elbows.

Second, the management team set up a new task force charged with incorporating best-practice templates for key tasks into the procedure manual. Epstein resolved that as soon as the new manual was available, project teams would be assessed and rewarded not only for their effectiveness in executing their projects but also for improvements to the templates that they suggested in their postproject reviews.

Finally, management asked the development organization to begin reporting, on a trial basis, the process data recommended by the task force. Epstein reasoned that he would need those data to gauge the effectiveness of the new process-management approach during the coming months.

First Results

During the early days of the task force, some managers and staff in ConnectCo's development organization had worried that process management would undermine the autonomy they needed. To people engaged in creative, nonrepetitive work, process models, detailed metrics, and process templates sounded like a recipe for regimentation and alienation.

By the time the task force made its recommendations, however, most people had begun

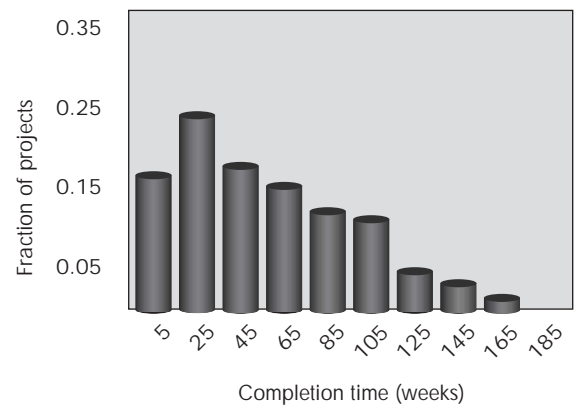## ConnectCo's Historical Project-Completion Times

New Products
Average completion time: 80 weeks
Worst 10% take longer than 140 weeks
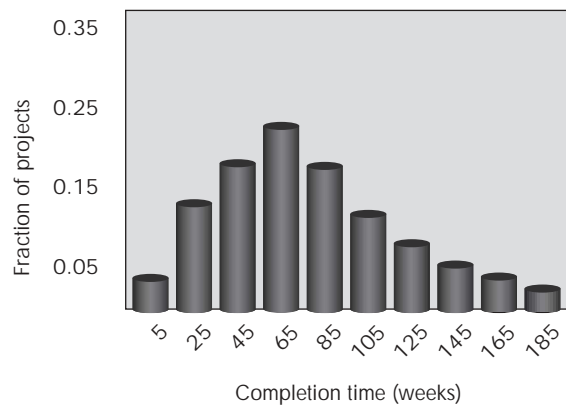
Product Extensions
Average completion time: 54 weeks
Worst 10% take longer than 100 weeks

to see process management as an exciting, new way to understand their work. After all, everyone cared about time to market. Moreover, the task force had involved colleagues in the process management effort, and Epstein had committed to using the new process measures for improving processes, not assigning blame.

During the next year, the company trimmed its project portfolio from 32 to 22 ongoing projects. As Epstein had predicted, many projects had been close to completion and could be wrapped up quickly. ConnectCo completed 18 projects that year, almost 30% more than its historical average.

The senior management team was firm in its commitment to take on fewer new projects. In the past, it had accepted projects based on their business attractiveness and then let them sit in the backlog. Now the team adopted a strict rule that no project could start until the required resources were available. As a result, ConnectCo accepted only eight new projects during that year, 60% of its historical average.

The new rules did generate some resistance. Marketing managers feared that strict limits on new projects would stymie their ability to respond to customer demands. Moreover, their bonuses were tied to the value of new contracts. Bill Shaw, the head of marketing, took the latter problem to his staff, and they came up with a new pay system that reduced bonuses in exchange for higher base salaries and established a broader set of performance goals for determining bonuses.

Like Shaw, Epstein was concerned that turning down too many requests from long-standing customers would weaken those relationships. Now that management had a better grasp of the development organization's capabilities, Epstein decided that in the coming year ConnectCo should take on 11 or so new projects and push for a goal of 16 ongoing projects.

The improved balance between resources and workload alleviated many stresses in the development organization. But some old habits died hard. The queues were indeed shorter, but project leaders were still eager to push their projects to the front of the line. One project in particular became something of a cause célèbre. The project manager, Claire Chen, was working with a customer who was under great time pressure. When Chen tried to accelerate the schedule by pleading with the engineers and technicians, they refused. She appealed to the senior management group and criticized the new approach as dangerously rigid.

In the interest of stabilizing the development process, Murphy's task force had encouraged departments to adopt a first-in, first-out approach to managing their in-boxes. The new plan provided no guidelines for dealing with real emergencies such as Chen's project. The senior managers decided that a refinement was in order: The rule against expediting projects was too rigid. Indeed, now that capacity utilization had been reduced, expedited projects
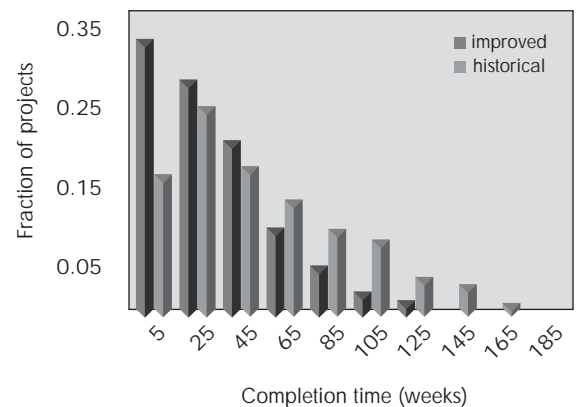
Estimated Improvements in Completion Times

New Products
Average completion time: 51 weeks
Worst 10% take longer than 85 weeks
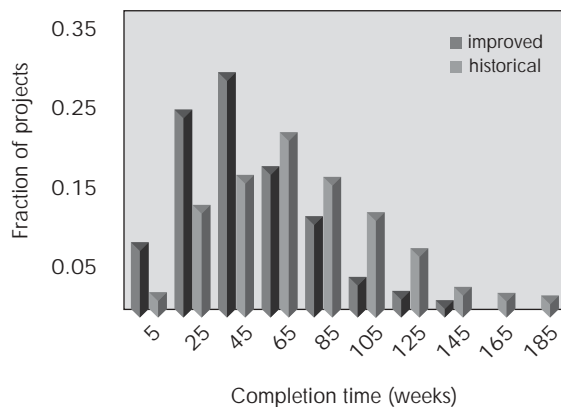
Product Extensions
Average completion time: 32 weeks
Worst 10% take longer than 60 weeks

The Process Reporting Form

A. Report for Each Project in Progress

	Concept Development	Prototyping	Market Plan	Manufacturing Scale-up	Specifications	Final Testing

Contributions
(hours spent this month on each task):
  Engineering
  Technical services
  Marketing
  Specifications
  Manufacturing engineering

Project calendar:
  Date task begun
  Date task completed

B. Summary Report for All Projects (This Month and Year to Date)

Projects

	New Products	Product Extensions

YTD project starts
YTD project completions
Number of projects in progress

Resources

	Engineering	Technical Services	Marketing	Specifications	Manufacturing Engineering

Full-time equivalents (FTEs)

Hours spent this month on:
  Support
  Administration
  Total new products
  Total product extensions

Processes

	Concept Development	Prototyping	Market Plan	Manufacturing Scale-up	Specifications	Final Testing

Efficiency
(time spent on each task):
  Mean of all projects completed this month
  Mean of all projects completed YTD
  90th percentile of all projects completed YTD

Yield
(number of iterations required by each task):
  Mean of all projects completed this month
  Mean of all projects completed YTD
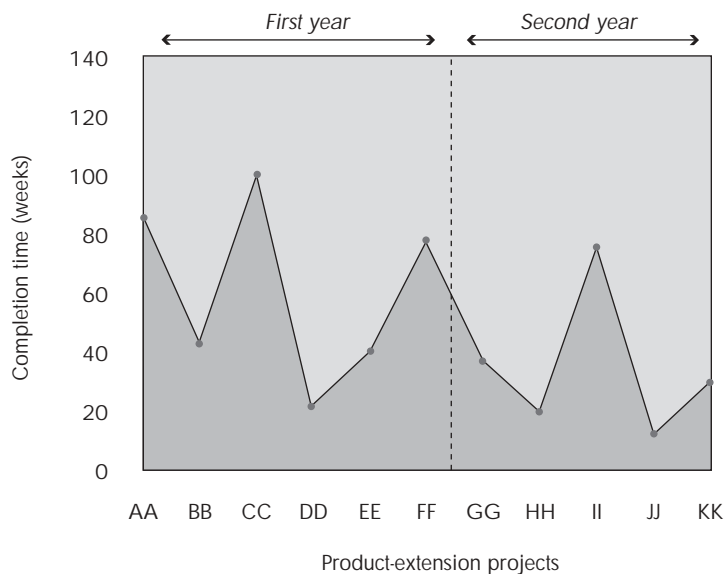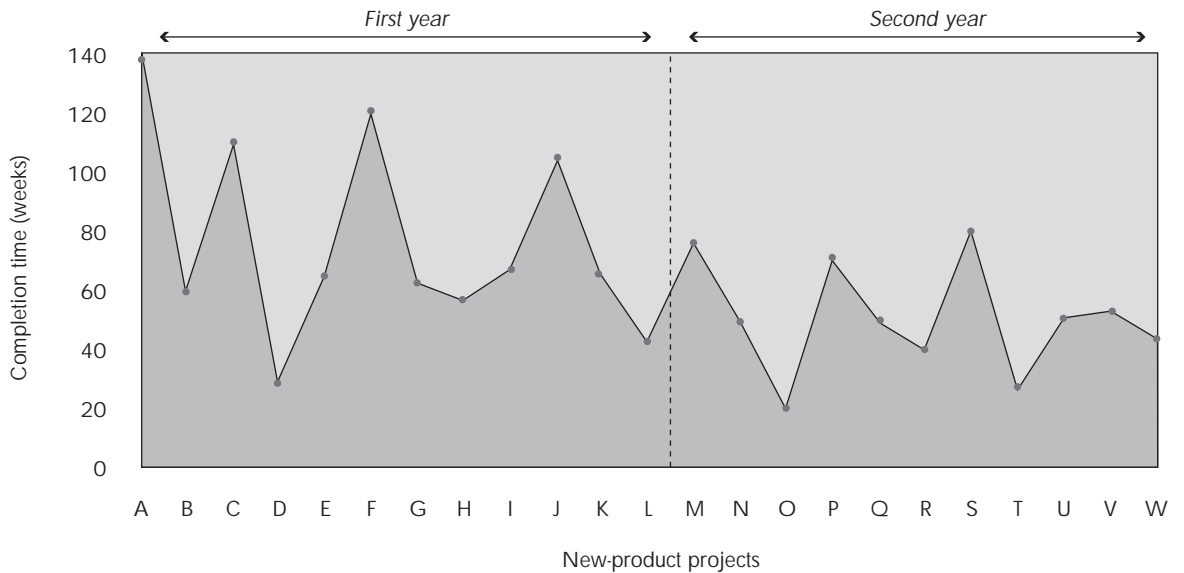 90th percentile of all projects completed YTD

would be less disruptive. But the senior managers had no desire to let the more aggressive project leaders again decide which projects received special treatment. They decided to allow some projects to be designated urgent but mandated that only the senior management team—not project leaders—could confer that status.

The program for training technicians went more smoothly. Most technicians were eager to expand their jobs. There were some rumblings, however, about the need for salary increases commensurate with the new responsibilities. Epstein decided that technicians with broader

skills did deserve higher pay. Some engineers were initially reluctant to relinquish their responsibility for test programming. But since the new division of labor freed up so much of their time, they quickly changed their minds.

This cross-functional-training effort also served as the pilot for creating best-practice templates. Because the engineers had been responsible for the test programming, Gilles asked them to develop a template for translating test parameters into specific test programs. A group of engineers laid out a generic programming process and identified five different testing scenarios that called for slightly differ-

ConnectCo's Results

First year / Second year

Completion time (weeks)

New-product projects

A B C D E F G H I J K L M N O P Q R S T U V W

First year / Second year

Completion time (weeks)

Product-extension projects

AA BB CC DD EE FF GG HH II JJ KK

ent approaches. Over the next three months, the technicians discovered many ambiguities and inconsistencies in the templates. A team of engineers and technicians revised the procedures and eventually produced a 60-page manual, which the technicians found useful. Soon the technicians were adding their own notes and improvements to the manual.

Although some projects were still taking longer than Epstein would have liked, the average development cycle time in the second year was 35% less than the average time before the initiative. (See the exhibit "ConnectCo's Results.") Just as the task force had predicted, there were fewer projects in limbo. And by the end of the two years, ConnectCo had only 17 projects under way, down from 32 at the start.

By helping people identify projects that deviated from the averages, the new measurement system helped them deepen their under-standing of the process. Unusually long or short projects became learning opportunities. Post-project evaluations now pinpointed hidden bottlenecks, skill shortages, and template inadequacies—and the associated improvement opportunities. The company discovered, for example, that some projects were held up for weeks until the plant found time to conduct trial runs. ConnectCo invested in a pilot line in the lab, which ended up saving an additional two months on the average project. Through process management, continuous improvement had come to product development. ▽

Reprint 96202; *Harvard Business Review* OnPoint 4880
To order, see the next page
or call 800-988-0886 or 617-783-7500
or go to www.hbr.org

Getting the Most out of Your Product Development Process

Further Reading

ARTICLES

Creativity Is Not Enough
by Theodore Levitt
Harvard Business Review
Republished August 2002
Product no. 1628

Levitt agrees that process management—an example of what he might call **organizational structure**—can actually support rather than stifle creativity. Applying process management to product development, he would argue, enables you to surmount an all-too-common problem: how to turn creative ideas into *profitable* innovations. Generating ideas (creativity) is one thing; putting them to work (innovation) is quite another. In many companies, great ideas kick around, unused, for years because no one assumed responsibility for converting big talk into bigger action.

Levitt offers several guidelines for improving the innovation process: 1) *Demand responsible presentation of ideas.* Whenever anyone suggests an idea, require him or her to include information on the associated costs, risks, manpower, time, and specific people required to carry it out. 2) *Encourage people to start implementing their ideas.* In large organizations especially, stability, structure, and heft make innovation less risky. New ideas may rock your big corporate boat, but they won't capsize it. 3) *Provide a home for irresponsibly creative people.* Some people simply can't handle implementation. Designate a specialized group whose sole function is to receive these individuals' ideas, work them out, and follow through on the implementation details.

How Process Enterprises Really Work
by Michael Hammer and Steven Stanton
Harvard Business Review
November–December 1999
Product no. 7893

Once you've applied process management to streamline your company's product development, you need to take the critical next step: building management structures that *support* your streamlined processes, transforming your organization into what Hammer and Stanton call a **process enterprise**.

Process enterprises replace turf and hierarchy battles with new approaches to leadership, performance measurement, compensation, and training—all focused on enhancing flexibility and efficiency. To craft a process enterprise, the authors recommend creating a new managerial position: the process owner. Each process owner takes end-to-end responsibility for a particular process—which includes authority over work and budgets. He or she designs the process, measures its performance, and trains the front-line workers who perform it.

Process owners must work differently with each other *and* with the front line. For example, they need to focus on teamwork, negotiate and collaborate, exert influence rather than formal authority, and coach and develop (rather than control) front-line employees.