Originally published in *The PDMA Handbook of New Product Development*, Second Edition, Kenneth B. Kahn, Editor, 2004, by John Wiley & Sons, Inc. ISBN 0-471-48524-1

CHAPTER TWELVE

ACCELERATED PRODUCT DEVELOPMENT: TECHNIQUES AND TRAPS

Preston G. Smith

12.1 Introduction

Many managers approach rapid product development with unrealistic, distorted, or unclear objectives, so they are disappointed with the results they achieve. This chapter clarifies objectives, frames the capabilities and skills needed, and discusses pitfalls that have misled many managers in their search for development speed.

Time-to-market is such an alluring goal that providers of various techniques often list faster development as one of their technique's benefits. They are partially correct, but their technique may not speed up the aspect of development that you need. Also, many of the other chapters in this book provide the basic ingredients of rapid development, but they must be adapted in certain ways to provide significant improvements in speed. The goal of this chapter is to enable you to become a "smart shopper" for accelerated development techniques and to read the rest of this book with greater sensitivity to the time-saving potential of the tools it provides.

12.2 Time-To-Market Is Not Free

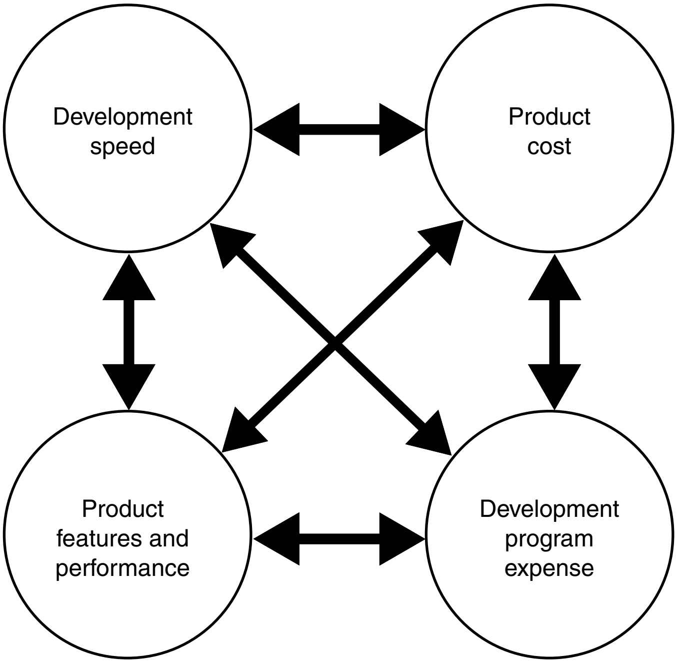
The first trap that awaits you is to assume that faster development is always desirable. Speed has its price, which might be paid by higher product cost, greater development expense, a reduced product feature set, higher risk, organizational changes and training, or staff burnout. Perhaps the easiest way to see this is to ask why you aren't moving faster now. If there were no impediment, you would already be taking advantage of speed techniques. You and your colleagues are smart, and most of the techniques are well known. Whether or not you have vocalized it, you know there is a price to be paid somewhere. Sometimes, the price is not worth benefit to be gained. Even if it is, if you continue to accelerate long enough, you will eventually reach this barrier.

The classic trade-offs of cycle time are against the first three items listed: higher product cost, greater development expense, and reduced product performance or feature set. These are illustrated in Figure 12.1. The six trade-offs pointed out here by the arrows can be calculated for each of your projects, as described in Chapter 2 of Smith and Reinertsen (1998). For instance, the most common trade-off is between cycle time and development expense, a value known as the cost of delay. The cost of delay may be less than $1000 per day, but for major projects in large companies, it can exceed $1 million per day in pretax profit. Using this value, you can wisely trade off time against money by hiring contractors, using airfreight liberally, and buying test equipment generously to cover peak loads. Likewise, using your project's trade-off value for time against the product feature set, your team can—on a cross-functional basis—prudently balance the features provided against the time available.

These classic trade-offs need not always be traded off against each other. For instance, you can gain on speed and development expense simultaneously, as shown in Figures 1-7 and 1-8 of Smith and Reinertsen (1998). However, in this case, you will pay in some other way, such as through process or organizational changes or in training. Also, beware of the "faster is cheaper" trap discussed at the end of the chapter.

Although you should be cautious about applying development speed where it isn't warranted, it is a valuable capability to possess. It is a potent arrow in your quiver for the projects that demand it. Besides, it is perhaps the best overall measure of the health of your product development capability. Think about sporting events: most of them are conducted as races against time, and for good reason. Anybody can walk, swim, or bicycle a kilometer, given enough time, but only the best can do it faster than anybody else. To win a race, everything must be in the best of condition: your skills, your weight, your mental attitude, your equipment, and the logistics supporting you—whether you are competing in sports or in the marketplace. If you can develop a product quickly, you probably can develop it to be superior in any other way you wish as well.

FIGURE 12.1 NEW DEVELOPMENT PROJECT TRADE-OFFS. EVERY DEVELOPMENT PROJECT REQUIRES TRADE-OFFS AMONG THESE FOUR ATTRIBUTES, AND YOU CAN CALCULATE THEM, EFFECTIVELY PUTTING CONVERSION VALUES ON EACH OF THE SIX ARROWS TO HELP YOU MAKE BETTER TRADE-OFF DECISIONS.



Source: Developing Products in Half the Time; Preston G. Smith and Donald G. Reinertsen; copyright © 1998 Preston G. Smith and Donald G. Reinertsen; this material is used by permission of John Wiley & Sons, Inc.

12.3 Types of Time-To-Market

Much of our confusion about improving cycle time stems from differing notions about how cycle time is measured. When does your cycle begin and when does it end? For example, one company did a wonderful job at cutting their cycle time in half, from the time they approved the project until the product was available for sale. However, they did this by bolstering front-end activities to reduce uncertainty later. When we added the extra front-end time back in, there was no net improvement. So, they decided that they needed to start measuring time earlier, when the product opportunity arose, and this changed their whole approach to shortening cycle time.

In addition, when organizations launch a cycle time initiative, they are seldom specific about the aspect of cycle time they wish to improve. To focus attention on suitable tools, the organization needs a clear business objective for speeding up development. Here is a sampling of the possibilities:

- *All-out speed:* This can be important in certain high-tech markets where products become outmoded quickly.
- *Minimizing schedule variation:* Many products depend on hitting a certain launch window, such as those that are seasonal or holiday related and those launched through an annual trade show. Predictability is more important than raw speed here.
- *Improving agility:* Many acceleration techniques also allow you to make changes, such as product features changes, more easily and later without disrupting the launch date. This is a potent weapon in a chaotic world.
- *Avoiding mistakes and rework:* Some delays stem from mistakes, so some managers concentrate on the mistakes to save not only time but also other wasted resources as well.
- *Improving productivity:* Although this is not actually a time-to-market objective, it is often the unspoken objective of managers who turn to accelerated development in order to obtain more new products from their resources or to cut expenses.
- *Overcoming sagging revenues or market share:* The thinking here is short-term and sales driven: if we could just get some new products out quickly now, we could overcome weak financial results. There is little long-term interest in improving business processes.
- *Sticking to schedule:* Many organizations exhibit poor schedule discipline, which means that development projects, meetings, and everything else simply happen whenever they happen. Basic schedule discipline will help them greatly.

None of these seven reasons is likely to fit your situation exactly. Please use them as a starting point to develop your objective in speeding up your development. Once you have created your rationale, share it with everyone in your organization. This will align everyone on techniques and approaches that will head your organization in this direction. Perhaps more importantly, it will diffuse potential rumors that can undercut your efforts, such as, “This is just a ploy to squeeze more work out of us” or, “Quality doesn’t matter anymore.

Let's skip non-value-adding steps such as voice-of-the-customer research or product reliability testing." (Such misinterpretations are covered later.)

12.4 Time-To-Market Tools and Techniques

This section is a survey of the tools, techniques, and approaches for shortening development cycle time. Many of these are covered in greater detail in other chapters, and the references at the end of this chapter provide added detail. The other chapters and references are often aimed at objectives other than cycle time, however, so please adjust the tools to gain significant cycle time benefit. For example, the first group below, Process Control Approaches, is often focused on ensuring that all steps of the process are completed with the objective of maximizing the product's marketability. This is a laudable goal, but you may find there are certain steps that contribute more to cycle time than to product marketability, so your objective may suggest cutting out or adjusting these steps. However, you do so in concert with other parts of the organization, realizing the costs and benefits of these steps. Furthermore, you may decide not to cut out these steps from your next project, because the costs and benefits associated with them differ for the next project.

An important and popular technique is redesigning your development process, mainly by overlapping activities; see Smith and Reinertsen (1998, Chapter 9) for details. In contrast, the other techniques listed below are often not so obvious.

12.4.1 Process Control Approaches

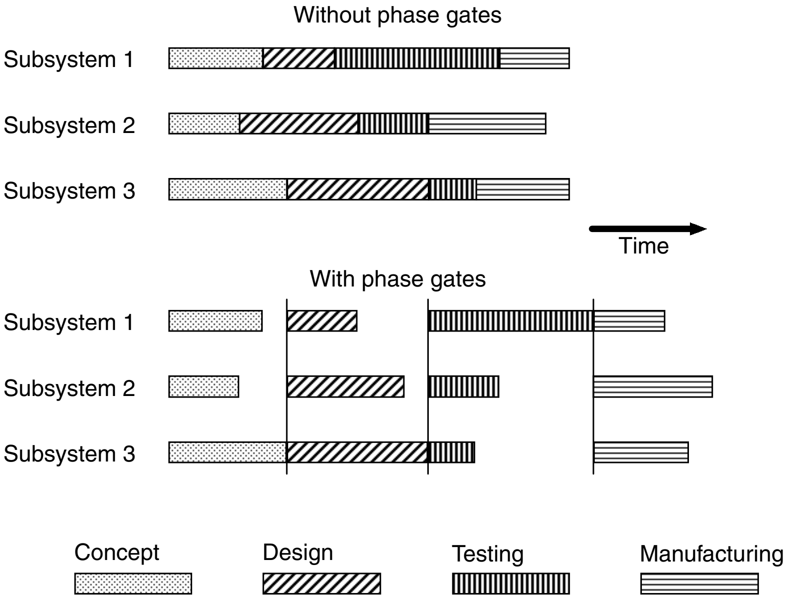
Most organizations that regularly develop products now have some sort of a phased development process, such as Stage-Gate®, Product Life Cycle, or Customer-to-Customer (C2C). These processes often claim to accelerate development, but their real emphasis is on consistency rather than speed. For example the primary reference on Stage-Gate® is Cooper (2001), whose subtitle is "Accelerating the Process from Idea to Launch." However, the objective of Cooper's book is primarily on creating a product that "wins" in the marketplace and secondly on aiming development resources at the organization's biggest opportunities. Speed plays a minor role.

Phased development processes divide development into sequential phases that allow management to check certain deliverables at the end of a phase and decide whether the project should be admitted to the next phase. This renders it easy for management to track progress, but speed suffers:

- The process is necessarily sequential.
- It is difficult or impossible to overlap activities between phases.
- Time is inevitably lost in completing reviews at gates (see Figure 12.2).
- The team is in limbo about the next phase until the last one has been approved.

The team being in limbo is an important point, because it sends the team mixed messages. Management has probably told the team that this project is urgent and that everything should be done to complete it quickly, yet management holds up the project until they decide whether the business wishes to continue with it. It is difficult to keep team momentum in high gear under such circumstances. The term *tollgate*, used by many firms for phased development systems,

FIGURE 12.2 EACH SUBSYSTEM HAS DIFFERENT TIMING NEEDS. WHEN PHASE GATES ARE USED, THE ORGANIZATION MUST WAIT FOR THE SLOWEST SUBSYSTEM AT EACH GATE. THERE ARE OTHER MORE SIGNIFICANT BUT LESS OBVIOUS DELAYS CONNECTED WITH PHASE GATES.



is apropos: stop and pay your toll before we will admit you to the next section of the turnpike.

There are ways of reducing these impediments to speed. Cooper (2001) proposes fuzzy gates. Note that fuzzy gates undermine the character of phased systems in that each phase is intended to be an escalating step of project investment, so management should independently decide at each gate whether they wish to continue the project investment. However, reality sides with fuzzy gates, because it is not practical to stop a project at a gate while awaiting a review. Will you lay off the team at the gate and hire a new one if you decide to fund the next phase? Momentum continues, but the team is hobbled—not a fast way to run a project.

An alternative is to announce that the project will continue at full speed presuming gate approval. But this is no different from having no gates, because management always has the choice to kill the project at any time. Also, you can minimize the impact of gate delays by having fewer gates. The project history of most organizations shows that, usually, projects are actually killed at only a couple of gates that involve particularly large resource commitments. Conclusion: consider these gate issues and design your process carefully to reflect what you wish to actually happen at your gates.

There are two ways of avoiding phased development processes. One is to shift the burden of controlling progress to the team. Phased processes are attempts to build judgment into the process rather than letting it stay with the team. A team that makes most of its own decisions can move much faster—and it usually makes much better decisions, because it has more accurate, fresher information from which to operate. One way of accomplishing this is by using development agreements between the team and management. As explained in Chapter 14 of Smith and Reinertsen (1998), these are essentially contracts between the team and management that specify the team's and management's authority and obligations. A similar and more recent approach is the bounding box, essentially a management by exceptions technique in which certain critical parameters of the project, such as profit margin, project budget, product performance level, and launch date, are negotiated as the bounding box. The team is free to move ahead unimpeded by phase boundaries as long as it stays within the box. Management regularly checks that the team remains within the box, and it is also the team's responsibility to notify management quickly if it finds that it is leaving the box. For more on bounding box, see Smith (2004).

Another way of avoiding phased development is to fundamentally change its sequential basis. Software development has done this by moving from its former waterfall (sequential) process to an inherently iterative one, as described by Kruchten (2000). Koen et al. (2002) found that a sequential model was inappropriate for front-end activities in product development, so they created a relationship model for these initial product development activities that clearly avoids sequential implications.

Eppinger (2001) has taken this one step further to show that product design is an inherently iterative process for all but the simplest products; that is, product design, by nature, is not sequential but instead depends on going back and revisiting earlier assumptions and adjusting them until the whole design is in balance. This occurs because there are points in the process where the designer does not have enough information to proceed and so must make some assumptions that are most likely partially wrong. This is no reflection on the designer but rather a characteristic inherent to innovation.

For more on product development processes, see Chapters 1, 4, 6, and 8.

12.4.2 Project Management Tools

Increasingly, product development is being viewed as a project to be managed. Project management has been reinforced in some companies by moving to program management, which brings more of a business perspective to the project. Other firms are formalizing project management through a project office structure, as covered by Englund, Graham, and Dinsmore (2003).

The basic project management skills of planning, scheduling, budgeting, and creating work breakdown structures are valuable for effective, fast product development. Certainly, the concept of critical path is essential to managing a fast-moving project, and Leach (2000) has extended critical path to the concept of critical chain.

As you apply project management to speed up product development, however, keep four things in mind:

- The project manager must have authority to take action. In some companies, the project manager is more of a clerk who knows exactly which activities are on the critical path but has no power to move them along. This is frustrating for the project manager and provides the organization with little time-to-market benefit. Managing an accelerated project is about people skills and empowerment as much as it is about the tools of scheduling, work breakdown structure, and the like.
- Project management provides effective tools for estimating and planning project schedules, but some marketing-driven organizations ignore these and dictate unrealistic schedules based on market pressures. This leads to disbelief, chaos, and weakened morale at the working level; in general, it back-

fires. In order for project management to be useful, you must believe in its tools.

- Employ project management software, but don't become subservient to it. Do not plan too far into the future, subdivide activities too finely, or spend time on the computer updating the project model when working directly with project staff would move the project ahead faster.
- Establish a baseline schedule and do not alter it. The temptation is great to update the baseline when the project is replanned, but then the project tends to keep slipping as the past is forgotten.

Related material appears in Chapter 10.

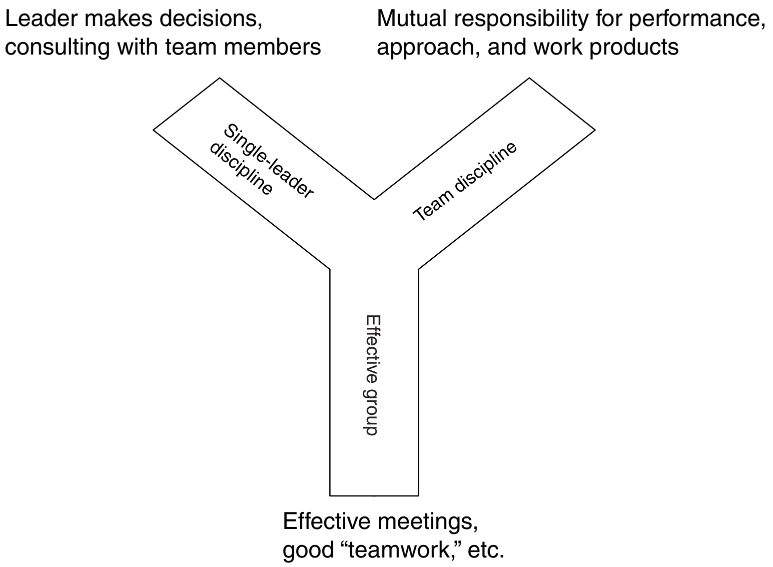
12.4.3 Team Techniques

A high-performance team is perhaps your most potent tool for accelerating a project, but high-performance teams are rare and demanding. Virtually all organizations developing products today would claim to employ a development team. However, most of these "teams" are what Katzenbach and Smith (2001) would call an effective group. An effective group has the basic skills to conduct effective meetings and respect members' contributions. But they are not knit together to achieve performance. There are two ways to achieve greater performance. One is called a single-leader discipline by Katzenbach and Smith (notice that they are quite restrictive about applying the term *team*, and would not apply it here). In this structure, the leader basically makes the decisions—normally after consulting with members—and remains accountable for them. In the team discipline, the team *jointly* makes decisions and produces work products. In the team discipline, a team member cannot fail; only the team can fail, since members are locked together by joint goals and work products. The team discipline takes more work to set up, and it can be an uncomfortable environment, since a member is accountable for others' shortcomings. See Figure 12.3 for elaboration of these three options.

Katzenbach and Smith suggest explicitly choosing the structure independently for each project, based on its requirements. The team discipline can greatly accelerate projects that can employ its strengths, but it has a higher setup investment. In general, product development projects gain from moving beyond the effective group, because they can benefit from joint work products.

However, recognize that product development projects, even within an organization, vary greatly in their need for team performance. Highly innovative breakthrough products and new platforms would benefit from investing in the team discipline, but an effective group could probably handle a repack-

FIGURE 12.3 THREE TEAM OPTIONS. FOR EACH PROJECT, EXPLICITLY CHOOSE THE ONE THAT PROVIDES THE BEST BLEND OF PERFORMANCE (SPEED) VERSUS SETUP EFFORT.


Leader makes decisions, consulting with team members
Mutual responsibility for performance, approach, and work products
Single-leader discipline
Team discipline
Effective group
Effective meetings, good "teamwork," etc.

Adapted from *The Discipline of Teams* by Jon R. Katzenbach and Douglas K. Smith; copyright © 2001 by Jon R. Katzenbach and Douglas K. Smith. Used with permission of Jon R. Katzenbach, Senior Partner of Katzenbach Partners LLC.

aging effort more efficiently. See Smith and Reinertsen (1998, pages 141–148) for more on tailoring your team structure to the project at hand.

Another major factor in the team's ability to move quickly is co-location. In the twentieth century, this meant having all critical members (engineering, marketing, manufacturing, and perhaps finance, procurement, regulatory, and others) physically close enough together to overhear each other's conversations. In the twenty-first century, with global organizations, global markets, development outsourcing and alliances, and new communication tools, many teams try to co-locate "virtually," with mixed results. Smith and Blanck (2002) discuss the pitfalls of dispersed teams and suggest means of improving their effectiveness. These include:

- Using as much partial co-location as possible, for example, by gathering at the start of the project, gathering all team members at each site close together, and identifying the heavy communication partners and co-locating them

- Deciding, as a team, on protocols to overcome the major delays involved in communicating at a distance, such as by specifying that an email will receive a reply within four hours, even if the answer must be incomplete
- Not depending on technology to overcome distance but only to be an enabler of enhanced people skills.

For more on teams, see Chapter 9.

12.4.4 Product-Related Tools

There is more to rapid development than process, organization, and project management. Many opportunities exist in the product itself. Developing a deep understanding of the customer and how the product is likely to be used is critical. This allows you to make quick but wise decisions in balancing the benefit offered by a feature against the time required to develop it. To do this, you must also know your project's economic trade-off rules, as shown in Figure 12.1. Moreover, a team that understands its customers will be able to reach such conclusions on the fly, rather than by consulting with the marketing department and conducting a new round of market research. In many cases, the project will be accelerated as a result, because you will conclude that a certain feature is no longer worth developing—at least not in this version of the product. In other words, rather than the common scope creep, you will enjoy wise scope shrinkage. For details, see Chapter 5 of Smith and Reinertsen (1998).

Another opportunity overlooked by many developers is the structure, or architecture, of the product and related opportunities to roll out feature sets incrementally rather than as a do-everything megaproject. Chapters 4 and 6 of Smith and Reinertsen (1998) describe the details.

Note that these tools are related: in order to effectively shrink the product scope on the fly and roll out incremental feature packages wisely, you will need a full-time marketing member co-located with your high-performance team, as suggested in the previous section.

12.4.5 Staffing Techniques

Too many projects proceed at half pace simply because they are half-staffed or staffing ramps up slowly, and often the worst bottlenecks are not in R&D. This is probably the most prevalent opportunity for improving cycle-time performance dramatically. Many firms are working on twice as many development projects as they could staff fully, which generally means that each project proceeds at half of the pace than it could if fully staffed. Often, management has a poor conception of its development capacity or the extent to which it is overloading the system. The medicine is simple but bitter: eliminate half of the projects for now, staff the remainder fully and finish them. Then—and only then—start some more. For a fuller explanation and for an illustration of what happens when you split people between projects, see Chapter 11 of Smith and Reinertsen (1998).

For more, see Chapters 1 and 9.

12.5 Time-To-Market Traps

As you proceed to accelerate your product development, you should be aware of some traps that await you so that you can respond to them wisely. Here are the main ones.

12.5.1 Scaring or Rewarding People

This area is a minefield. Many managers try to motivate the team with an aggressive time-to-market mandate. Although Swink (2002) shows that an explicit project objective is a critical success factor—and many teams have no project objective—to be effective, the objective must be ambitious while also being seen as achievable and compelling *by the team. Compelling* means that it is an outside milestone that everyone appreciates as being critical to the business.

Rewards for the team or for individuals can be effective, but just as often they backfire. See Smith and Reinertsen (1998:135–138) for details. Perhaps surprisingly, Swink's research (2003) shows that rewards for speed generally *slow down* development projects.

12.5.2 Skipping Steps

Many of us were well trained as children that "haste makes waste." From this, we jump to the conclusion that rapid development can be accomplished only by skipping its time-consuming steps, as Crawford (1992) discusses. An effective development process is scalable, so that it can be adjusted to the needs of each project by deleting steps if their total cost outweighs their benefit. But skipping steps arbitrarily without considering their total cost is not the route to effective development, and it will not save time in the end.

12.5.3 Process Eliminates Mistakes

Some organizations try to build a watertight development process that will eliminate mistakes, and they thereby hamstring their development teams, preventing the obvious mistakes but also most opportunities to move quickly—and even to innovate outside of narrow boundaries. You will be much faster, while catching a more robust variety of mistakes, by empowering the team to run its project, as discussed earlier under "Process Control Approaches."

12.5.4 Product Development Is Engineering

Because the majority of the labor that goes into product development is normally technical, there is a great tendency to relegate product development to engineering. However, most of the opportunities to cut cycle time are cross-functional in nature, for example, scope creep problems. If you really want to accelerate development, look at the interdepartmental interfaces first.

12.5.5 Faster Is Cheaper

The beginning of this chapter emphasizes that rapid development is not free—you will pay for it in one way or another. Many managers blissfully assume that if they can develop a product in half the time, their labor expense will diminish commensurately. They do not comprehend that one of the most powerful tools in rapid development is fully staffing the team with full-time participants, which will roughly double their labor burn rate, although it will prevail for only half as long. In addition, a co-located team and a team following the team discipline will generally incur extra start-up costs and require higher-quality (more expensive) staff and more training.

Furthermore, even if these premiums did not exist, it is dangerous to assume that faster means cheaper. Under this assumption, management will starve the team for the resources needed to move to faster modes, and they will never see the acceleration they desire. It is much better to be relatively lavish in the beginning and then cut back after you have demonstrated success.

12.5.6 Outcome Outweighs Approach

Some hard-pressed managers turn to rapid development as a panacea to bring one badly needed new product to market sooner. They succeed at this, but because they have concentrated on the destination, they haven't noticed the route they have taken, and they cannot repeat it. If you want to build a repeatable rapid development capability, your first product through this approach must be subservient to developing the approach.

A large part of rapid development is in developing certain skills by team members and management as they proceed. These skills are a valuable, scarce asset. If you do not focus on developing skills, you will have nothing at the end besides your new product.

12.5.7 Technology Accelerates Development

Speeding up development is hard work. Some purveyors of technologies suggest that their technologies will reduce your development time. Technologies, such as computer networking, videoconferencing, product data management software, and computer-aided design (CAD), can support an improvement, but the core of the change must be in management styles, team makeup and behavior, and similar items that cannot be purchased and installed so easily. Start shopping for technology *after* you have chosen the management tools you plan how to implement projects. Also, allow extra time in your first couple of projects that use the new technology to learn it and work the kinks out of it.

12.6 Conclusion

Although you have many large opportunities to accelerate development, achieving results will require persistent hard work. Many different tools are available to you, and few of these are esoteric or obscure. The tools generally reinforce each other, so plan to combine them. Freely adapt them to your objectives, strengths, weaknesses, resources, and markets. Finally, recognize and celebrate the value of the asset you are creating, because this will fuel further progress.

References

Cooper, Robert G., *Winning at New Products.* Cambridge, MA: Perseus Publishing, 2001.

Crawford, Merle C., "The Hidden Costs of Accelerated Product Development," *Journal of Product Innovation Management,* 9(3):188–199 (September 1992).

Englund, Randall L., Graham, Robert J., and Dinsmore, Paul C. *Creating the Project Management Office.* San Francisco: Jossey-Bass, 2003.

Eppinger Steven D., "Innovation at the Speed of Information," *Harvard Business Review* 79(1): 149–158 (January, 2001).

Katzenbach, Jon R. and Smith, Douglas K., *The Discipline of Teams*. New York: John Wiley & Sons, Inc., 2001.

Koen, P., et al., "Fuzzy Front End: Effective Methods, Tools, and Techniques," in P. Belliveau, A. Griffin, and S. Somermeyer, eds. *The PDMA Toolbook for New Product Development.* New York: John Wiley & Sons, Inc., 2002.

Kruchten, Philippe. *The Rational Unified Process,* 2nd Ed. Reading MA: Addison-Wesley, 2002.

Leach, Lawrence P., *Critical Chain Project Management.* Norwood, MA: Artech House, 2000.

Smith, Preston G., "Concurrent Product Development Teams" in D. I. Cleland, ed. *Field Guide to Project Management*, 2nd Edition. New York: John Wiley & Sons, Inc., 2004.

Smith, Preston G. and Blanck, Emily L., "From Experience: Leading Dispersed Teams," *Journal of Product Innovation Management*, 19(4): 294–304 (July 2004).

Smith, Preston G. and Reinertsen, Donald G. *Developing Products in Half the Time*. New York: John Wiley & Sons, Inc., 1998.

Swink, Morgan. "Product Development—Faster, On-time," *Research-Technology Management*, 45(4): 50–58 (July–August 2002)

Swink, Morgan. "Completing Projects On-Time: How Project Acceleration Affects New Product Development," *Journal of Engineering and Technology Management*, 20(4): 319–344 (December 2003).

Preston G. Smith, as a principal with the consultancy New Product Dynamics, has specialized in accelerated product development for 20 years, and he has helped dozens of companies in 23 countries to adopt the techniques covered in this chapter. His book, *Developing Products in Half the Time*, and his article, "Reaping Benefit from Speed to Market," have been instrumental in helping many firms to bring their new products to market faster and more effectively. Preston has also served 20 years as an engineer and manager in both small and large companies. He holds an engineering Ph.D. from Stanford University and is a Certified Management Consultant.