Implementing the Principles of the Toyota Development System — Myths and Realities

by Michael N. Kennedy

Seeing the Potential

What makes the Toyota Development System (TDS) so powerful and unique? Start by looking at the results. In half the time of their competitors, Toyota develops cars widely acclaimed for the highest quality and reliability in the industry. Toyota is consistently far more profitable than all other automakers. Toyota's product development times and costs are 2-3 X shorter than Western, and development risks are much lower. Looking deeper, as outlined in Figure 1, the characteristics of TDS differ significantly from those prevalent in western companies.

My own work history provides a dramatic contrast. Prior to my conversion to the Toyota way of thinking, I worked 33 years for Texas Instruments Defense Systems (TI) in various engineering and management positions, both in development and manufacturing. By all accounts, TI was a leader in development capability. We won the Malcolm Baldrige National Quality Award; we were highly benchmarked for Six Sigma, for our business process engineering capability, for our Integrated Product Development process, and for our excellence in deploying CAD/CAM design automation. We were even benchmarked for our effectiveness in benchmarking. During my last ten years at TI, I was personally and proudly involved in all of these improvement initiatives.

However, there was a problem. In spite of all of these accolades, TI was actually degrading in many important areas. Our product costs were rising; our scheduling performance was deteriorating; our overhead rates were increasing; and our engineering productivity rate was measured at only about 20 percent. However,

In Brief

Toyota's product development system (TDS) is as unique as TPS, but harder to comprehend, and arguably more important for the ongoing success of a company. TDS principles seem diametrically opposed to many "best practice" concepts advocated as the cornerstone for American firms' product development. Going for Toyota's level of development success re-creates the product development work culture.

Comparison of Toyota Development System with Western Systems

Characteristics	Western	TDS
Engineering productivity	20-40%	80% or greater
Project management	Administrative (Focused on compliance)	Technical (Focused on targets)
Concurrent engineering	Advisor role / DFM	Joint development
Learning between projects	Very little	Almost all
Schedule adherence	Very low	Almost always
Manufacturing problems	Systemic	Minor

Figure 1.

worst of all, there was no indication that our ability to capture and reuse product knowledge was improving and in reality, was quite abysmal.

My conversion was based simply on two aspects of the Toyota system. The first is that Toyota developers, and the managers, operate at 80 percent productivity and the second is their ability to successfully capture and transfer robust technical knowledge across projects and functional boundaries. This, coupled with their continued ability to provide great products, always on schedule, and with great profitability was too compelling to ignore.

Over the last few years, I have visited many companies. Their similarities with TI have been amazing. All have similar product development processes, similar approaches for project reviews, similar management and engineering capabilities, and similar improvement initiatives for their product development process — but they also have similar low productivity rates and minimal real learning across projects. I also found that despite very little disagreement with the excellence of the Toyota principles, very few companies have made significant progress toward achieving those principles. Why?

The Essence of the Change

So how does Toyota achieve their level of excellence through TDS? For detailed understanding of the principles, I refer you to my book.[1] In addition, a number of articles written through the University of Michigan describe these concepts and the Toyota way of thinking. This article only sketches the pertinent principles and concepts so as to better understand the challenge of the change.

There are five unique, fundamental principles of the Toyota system. First, and most important, is the rigorous thought process by which *everyone* approaches development. Dr. Allen Ward, my mentor who investigated Toyota's practices for many years, called it the LAMDA cycle — everyone is responsible for ***L****ooking* at the problem; ***A****sking* why (the five whys); ***M****odeling* potential solutions, ***D****iscussing* the details, and ***A****cting* on the results, and then repeating the cycle as necessary. This thought process fundamentally opposes the predominant theory of scientific management in which only *experts* are primarily responsible for the technical excellence of the design process.

The other four principles are 1) *Set-based Engineering* that allows many alternatives to be simultaneously evaluated early in the project; 2) *Customer-Focused Entrepreneurial Leadership* that provides the required technical leadership for design convergence; 3) *Workforce Expertise* that provides the core development capability, and 4) a *Pull-based Planning and Control System* focused on achieving visible results rather than executing tasks.

However, the real magic of the Toyota system is that all of these principles combine into an incredible learning environment. Products continually emerge from this interaction of knowledge having been fully tested at the initial stages of each new project, and learning is robustly transferable to all future projects. In effect, they have developed a language of knowledge and a system that aggressively creates usable knowledge, maintains its quality, and leverages its use for all future projects.

Fundamental change is necessary to emulate the TDS. The development culture must move from scientific management principles to rapid, pervasive organizational learning focused on the needs of the customer. The techniques range from developing data banks of cost-technology trade-off curves to selecting the best design options at the last possible time. None of these techniques is fully effective unless adopters fully understand that they must further the continuous improvement of knowledge before they can improve product design processes.

The Myths of Change

Over the past few years, many companies have been attempting to transform themselves to achieve the development culture at Toyota, but with minimal success. I believe the difficulty is largely due to focusing on the wrong aspects of the TDS. Rather than learning the principles so as to truly transform their organizations, the companies tend to focus on copying attributes that are not essential to success. The following all-too-common myths are the wrong things to learn from the TDS.

Copy the practices of Toyota

The logic is that Toyota has spent 50 years evolving their techniques, so just copy them. That is exactly the reason *not* to do so. The Toyota system evolved from a different culture and mostly in a different time, and is continuing to evolve. Trying to insert a snapshot of these techniques into a different corporate culture now will almost surely result in cultural shock while still missing the real essence of the differences. Even if successful, the best the company can achieve is the capability that Toyota has already surpassed.

Information technology is not important

In many ways, this is a corollary of the first myth. Toyota's learning based system was built on a manual system of knowledge management. Toyota has long believed that excessive computer automation actually detracts from the personal knowledge of the developers. Yet, while Toyota has always maintained healthy skepticism about information technology (IT), it has never been technophobic. Rather, Toyota is selective and careful in its adoption: where IT makes no sense (such as ERP systems that are antithetical to pull production principles) Toyota steers away; where IT makes sense, Toyota experiments carefully. Toyota Powertrain is presently well along in evolving its approach to Product Lifecycle Management (PLM) technology, following its own path and driving its vendors' development instead of the other way around. I cannot imagine a modern adaptation of the Toyota principles — and that includes Toyota's own adaptation — not including a carefully constructed, but heavy dose, of enabling computer technology.

Standard process definitions provide a mechanism to enforce the Toyota way

Over time, most large companies have developed fairly complex standard product development processes. They are generally well-documented and have strong ownership, although from my experience, they are not as widely accepted and utilized as the owners tend to believe. The fact that so many companies insist on this level of structure is a monument to the scientific theory of management where experts define the work standards. The first reaction of a company adopting the Toyota learning paradigm is to modify their process standards to embed the Toyota principles, and thereby deploy those principles to all groups who use the new standard processes. While I can't say with certainty this won't work, I suggest caution because most standard product development processes are focused on task compliance, not learning efficiency. Toyota would tend to recommend extreme simplicity in those standards, and move as much focus as possible away from process detail and on to product knowledge.

Lean manufacturing techniques can lead the transformation in development

The continuous removal of waste is a mainstay for implementing the principles of lean production. The techniques of Value Stream Mapping (VSM) are well understood and practiced effectively within most major manufacturing suppliers. It is a very appealing approach to simply migrate those techniques into the development environment and systematically remove developmental waste. I will not argue the fact that any waste removal is positive and in lieu of a broader goal, will significantly lean out the development processes. However, this approach will not create the learning-based development environment of Toyota. VSM is a very powerful continuous improvement methodology, but will not fundamentally reengineer the cultural underpinnings of the development environment.

Gate reviews should be the basis of project status

Toyota has design reviews at distinct milestones, and these tend to resemble gate reviews to the many in the West that are practicing the popular gate review methods. However, in the world of scientific management, gate reviews focus on compliance to the standards, or at least apparent compliance, ensuring that all the necessary decisions have been made and that all the work based on those design decisions has been completed. Thus, the earlier the decisions are made, the easier it is to get that work done before the gate review. This is the opposite of the effect that Toyota is looking for. A Toyota chief engineer was quoted asserting that his primary role was to force design decisions as late as possible. In a learning environment, design reviews focus on the knowledge that must be acquired in order to make a decision. The decision has not yet been made; the design review is for making that decision. The Toyota design reviews are to ensure that the knowledge that has been acquired truly justifies the decision that is being made. Thus, rather than promoting premature decisions (making it easier for people to meet the gate reviews on time), the Toyota design reviews prevent premature decisions.

Task-based planning and control systems are necessary

With task-based project management so prevalent in companies, the natural first step in migration to Toyota practices is to define the Toyota-like tasks and put them into the task-based project management tool so that you can manage it. If however, the Toyota development approach is to continually adjust the actions, and even specifications, based on the results of what is being learned during the development work; then any attempt to pre-define the tasks that must be performed during a project is a waste — or worse, an impediment. A pull and responsibility-based system, one of the Toyota principles, is much better aligned for this type of learning environment.

Successful cultural change can be led by experts

The traditional approach for implementing major corporate change follows the scientific management model: Experts define the changes, pilot the changes in a controlled environment, and then lead the rollout through a process of training and convincing. This approach can be effective up to a point in manufacturing and other transactional processes, but has not proven effective for major cultural change in innovative or development processes. The experts just don't know enough and aren't respected enough to work through all the critical details and interaction. Even if useful change is instigated, where will the continuous improvement — experiential self-learning — come from when the experts depart? In my opinion, major cultural changes can only be successful by establishing a clear change vision and engaging the workforce, the true experts in their jobs, to define the process and organizational details.

The Opportunity for Real Change (How you can build from Toyota)

Now that I have described what won't work, what will? Rather than explain and illustrate the key Toyota principles (which you can get from my book or from numerous articles written by Dr. Allen Ward and others at the University of Michigan), this section will focus on how to change properly. If done correctly, this change is as close to "no-risk" as a major cultural change can be. If Toyota achieves 80 percent customer-focused productivity while retaining almost all relevant product knowledge and if your company is at 20 percent or even 40 percent productivity with little retention of technical knowledge, then the real risk is doing nothing. Any movement improving these capabilities will be positive.

At Toyota, the real importance of Set-Based Engineering is that it enables early inexpensive testing of relevant product data with sufficient robustness to drive their ability to "test and design" as opposed to "design and test." The key is that the early testing provides rich, robust, and enduring technical knowledge. Achieving this ability needs to be a major requirement of the change at any company.

As a result, the change goal should be "*the rapid generation and the continuous flow of robust technical knowledge across all generations of products.*" This is a huge difference from seeing the goal as the efficient flow of work to define the product. In manufacturing, continuous high quality material flow across the entire supply chain represents a lean environment. Similarly, the continuous flow of robust technical knowledge into an ongoing stream of excellent products represents a lean development environment.

A four-phase process for making the change seems to work. This change should start small — subassembly families, or other subsets, and progress project-by-project. However, it is critical that change momentum be generated and maintained. Within a year, before interest dies, progress should be evident, completing all steps on selected product assemblies. Expanding quicker by overlapping phases is encouraged as confidence grows in the process and results. The phases are:

- Understand the underlying physics that drives correct product decisions
- Capture the physics-based knowledge in reusable form
- Learn to design based on that knowledge
- Learn to expand that knowledge into reusable sets of solutions.

The initial phase is to understand the core knowledge that drives customers' needs at the lowest possible level of design, and to understand the physics-based design parameters that drive those needs. This will allow the robust knowledge to be understood, collected, and the establishment of standards for effective communication of that knowledge. At Toyota, this is done through simple standardized problem solving methodologies, including most importantly standardized trade-off curves representing the core product knowledge.

I suggest starting with a subassembly family, or better yet a subset of a product family. The approach is to understand the customer's true requirements and map them rigorously to the design variables that impact them. Follow this by identifying the trade-off relationships and the logic for using them effectively to make design decisions. This process will prove feasibility, and begin setting standards to define the "language of knowledge" required for using core technical knowledge as the basis for development.

The second phase is capturing existing knowledge in the context of the newly established knowledge standards. This knowledge will rarely be found in existing PDM or PLM systems or on product drawings. It is in the heads or file cabinets of your best design and manufacturing engineers. I expect that much of this knowledge will not have verifiable supporting test data. An example is the engineer that can make a good off-the top-of-the-head estimate of the size radiator required for a specific engine. Capturing the intuitive knowledge of your best technical personnel into usable trade-off curves and logic is a huge step.

The third phase is to begin designing using the new language of knowledge. The ability to design and make decisions based on the underlying physics and trade-off relationships, as opposed to geometric relationships, must be learned, and it can be done within the confines of existing project constraints. In other words, a subassembly design team can define and test reusable knowledge even in a traditional point-based design environment. Despite the limitations that environment imposes, significant value will be gained from this phase, and it is an important step on the way to the broader implementation of true set-based design.

The fourth phase is implementing set-based development, which not only implies the use of the language of knowledge as the basis for development, but also implies the use of broader (and possibly conflicting) targets across all subassemblies, and a convergence process for negotiating and continually eliminating weak alternatives. It is certainly possible (and recommended) to expand set-based thinking across a few subassemblies at a time, as opposed to starting with an entire project.

The Bottom Line

I have no doubt that Toyota has pointed us in the right direction for quantum improvements in product development performance through the focus on knowledge. Unfortunately, most companies are rapidly moving in exactly the wrong direction by continuing to focus on compliance and structure. I encourage business development leaders to pause to understand the lessons of the Toyota Development System. Get involved by asking the right questions. Who is taking you where and why? Weigh the real risks and costs of changing direction, and act accordingly.

Michael N. Kennedy was the lead engineer on many projects during 33 years at Texas Instruments. For several years he collaborated with the late Dr. Allen Ward of the University of Michigan studying the Toyota Development System. He now works closely with the National Center for Manufacturing Sciences helping companies adopt this system.

Footnote:

1. Michael N. Kennedy, *Product Development for the Lean Enterprise*, The Oaklea Press, Richmond, VA, 2003. (Reviewed in *Target*, Second Issue, 2004, pp. 54-55.)