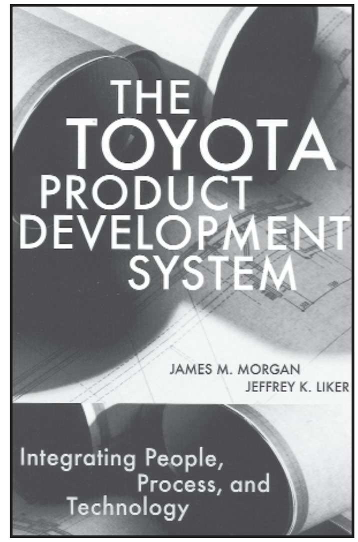
The Toyota Product Development System

James M. Morgan and Jeffrey K. Liker
Productivity Press, New York, 2006, 363 pp.
212/686-5900



This appears to be the most comprehensive book to date on the Toyota Product Development System (TPDS). Morgan and Liker call it Lean Product Development System, but it is really Toyota's, not American lean. It is the result of lengthy research at the University of Michigan headed by the late Alan Ward, to whom the authors give great credit. All the examples compare vehicle development between Toyota and North American car companies, but by careful reading and reflection, anyone interested in product development can learn how the principles apply to them. The book illustrates 13 principles of lean product development.

No amount of operational kaizen can overcome the waste locked up in a poor design. The intended customers may not be pleased either. Toyota's expertise developing new products is well recognized. Although not known for styling pizzazz, Toyota has fewer problems with recalls than competitors, and its leadtime for a derivative model designed on an existing platform has been as low as ten months. Its average is 12-15 months compared with 24 months in North America. The North American companies' leadtime is down from 36-40 months in the 1990s, so everyone is progressing, and shorter leadtimes mean more new models. Since 2003, the automotive industry has annually launched 60 or more new models each year into the North American market alone. No wonder you can't identify all the cars anymore.

The Toyota Product Development System, interacting with its vaunted production system, has become a big advantage to Toyota, because productivity is only one factor in overall waste. For instance, the 2006 Harbour Report shows several other assembly plants in North America to be more productive than Toyota's, but Ford's Atlanta Taurus Assembly, topping Harbour's list, is also on Ford's closure list. By designing quality in, then building it in, plus creating flexibility to change model mixes while minimizing spending on tools and equipment, Toyota's overall advantages offset small differences in assembly productivity.

Within Toyota, experience with the production system remains bedrock training for almost everyone having other responsibilities. For example, every new Toyota engineer does a stint in production — and at a dealership, probably selling vehicles. To design them to be made and sold, you need first-hand experience with the processes you are designing for. However, Toyota's greatest emphasis with engineers' career development is technical depth. They want "T-shaped" people, engineers with great technical expertise, but also having the breadth for easy collaboration among many functions as a normal part of daily work developing vehicles.

Well-known is that Toyota product development relies on a chief engineer (CE) for each car, A3 papers codifying technical knowledge, and selection of designs from tested data using trade-off curves, with designs jelled at the last possible time from among many alternatives. The chief engineer orchestrates a vehicle's design, and is responsible for design decisions. That, in a nutshell, is set-based design engineering.

A few benefits of TPDS can be derived by copying its techniques, but the entire system defies quick, easy emulation. It is based on a different concept of product design and a different organizational philosophy. The system developed over decades. It is still evolving. A few HR tweaks plus adopt-a-technique won't quickly replicate a system that can fully compete with it.

For instance, Toyota has invested years in developing each engineer, technically and otherwise, in their system to continuously improve collective knowledge, and in their processes for collaboration.

Assuming that turnover will be very low, it's a people-centered system. The discipline of the A3 papers and the mentoring of senior engineers develop tacit knowledge. Aided by the technical archives and process checklists, this base of experience prompts someone to ask the right questions throughout a design process. By taking time up front to review sets of tested design alternatives, most of the time late in the process is value-added work to launch the vehicle, not find-and-fix. Rework and engineering changes are minimal because the system stimulates continuous improvement of knowledge and skills. Attention to that, despite the pressures of immediate projects, lets every product development start further down the learning curve — more new learning, less re-learning.

For those ambitious to overhaul their product development processes, a brief review cannot convey the lessons of this book. A second or third re-read of some of the writing is worthwhile. Many of these ideas have to be digested slowly because TPDS is an interlocked system, not a collection of disconnected techniques.

A few Japanese terms may be new; *kentou*, for instance, the front-end study phase of new product development. So many people are involved in *kentou* that most project "doers" become familiar with a project and its customer needs at this stage. They begin thinking through knotty problems before project execution begins. That's well before North American lean companies start to refine a design using 3P.

Another term is *mizen bouchi*, meaning designed-in quality. This label covers a host of techniques that over time add up to continuous improvement of product design and process quality, becoming embedded in the culture — how things get done at Toyota.

Although hard to fully appreciate without explanatory examples, a quick summary of the 13 principles constituting the framework on which Morgan and Liker hang TPDS give a flavor for the book:

1. Establish customer-defined value to separate value-added from waste.
2. Front-load the product design process while there is design space to thoroughly explore alter native solutions.
3. Create a leveled product development process flow.
4. Use rigorous standardization to reduce variation, thereby creating both flexibility and predictable outcomes.
5. Develop a chief engineer system to integrate development from start to finish.
6. Organize to balance functional expertise and cross-functional integration (more is involved here than your father's matrix management).
7. Develop towering technical competence in all engineers.
8. Fully integrate suppliers into the product development system.
9. Build in learning and continuous improvement.
10. Build a culture to support excellence and relentless improvement.
11. Adapt technology to fit your people and your processes.
12. Align your organization through simple, visual communication.
13. Use powerful tools for standardization and organizational learning.

Reviewed by Robert W. Hall, editor-in-chief, Target *Magazine.*