ENTERING THE TIGER'S CAVE

Perspectives on Japanese and
Swedish Product Development

ENTERING THE TIGER'S CAVE

– Perspectives on Japanese and Swedish Product Development

DAG BERGSJÖ, Editor

ENTERING THE TIGER'S CAVE
Perspectives on Japanese and Swedish Product Development
Dag Bergsjö, Editor
© DAG BERGSJÖ
ISSN 1652-9243
Report no: 53

Published and Distributed by
Chalmers University of Technology
Department of Product and Production Development
Division of Product Development
SE – 412 96 Göteborg, Sweden

Telephone +46 (0)31 – 772 1000
URL: www.chalmers.se

Cover: Illustration by Christoffer Levandowski
Photo by: Anders Forslund

Printed in Sweden by
Chalmers Reproservice
Göteborg, 2011

FOREWORD

In a world that is getting smaller and where globalism is a fact, the need to understand and communicate with researchers and engineers from different cultural backgrounds becomes very important. As a researcher, it is your task to constantly question things and to find better solutions to problems. From the external viewer's point of view, Japan has been seen as a great success story, and the Toyota production system has been exported to, and been inspiration for, many companies outside of Japan. Influences from other cultures and the solutions that can be obtained from having a slightly different perspective on the world can be worth a great deal to researchers in product development. It is with this background and the desire to learn more about product development that we chose to create a course in international product development and to take on a trip to Japan to explore their culture and specifically their view of product development.

The planning of the course and the trip took about one year to complete. We wanted to visit as many Japanese industries and research centers as possible, and we also wanted to learn about the culture and the history of Japan and its success in product development before we got there. The first workshop was set up in Gothenburg, Sweden, for three days in 2010. This was followed up by a workshop in and around Stockholm a few months later. We were given the opportunity to learn more about Japanese product development methods, often referred to as Lean Product Development, and also the history, culture and language of Japan. Company visits in Gothenburg together with previous industrial experience gave us a basis for comparison of the actual product development.

When we arrived in Japan on the first of November 2010, we had a full program for the following two weeks. The days were spent in and around Tokyo, Nagoya, Kyoto and Osaka. Many of us also took the opportunity to use the excellent Japanese train system and our rail passes to really see the country. Some of us got to see the mountains around Nagano, the beaches of Shimoda and the beautiful islands outside Hiroshima. This was a welcome break from all the lectures and visits, and was time very well spent.

We got to learn a significant amount about the Japanese view on product development and production. The viewpoint of the customer as your god left a big imprint on my mind, as well as the principles of quality first (then we do the rest) and the continuous hunt for improvement and perfection. Simply put, we learned a lot from out visit, above all that we have a lot more to learn from each other. The old Japanese saying *Koketsu ni irazunba koji wo ezu* (If you do not enter the tiger's cave, you will not catch its cub) says that to be able to achieve you must go where no one else has gone. This was an effort in doing just that.

Gothenburg, Sweden, April 2011
Dag Bergsjö, editor

ACKNOWLEDGEMENT

We would like to gratefully thank all the participating companies and universities that contributed to the richness of this course and voyage.

First of all we would like to thank ProViking research school for financing the course, and Göran Gustafson for taking the initiative to undertake this project. We are also very thankful towards Professor Tomohiko Sakao from Linköping University for his support and guidance in Japan.

Further, we would like to acknowledge the support of all the companies, universities, and lecturers that made this course and study trip a great success. A special thanks, goes to:

- Autoliv
- BT products AB
- Professor Fujimoto at Tokyo University
- IKEA Service Office Japan
- Institute for Japanese Studies
- IVF Swerea
- Mr. Kasai, IDEA
- Professor Kimura at Hosei University
- Komatsu
- Scania CV
- Swedish Embassy in Tokyo
- Toyota Material Handling Group in TICO
- Toyota Motor Corporation, and
- Professor Ueno, Fellow of JST & Guest professor of Kanazawa Institute of technology

TABLE OF CONTENTS

PRODUCT DEVELOPMENT MANAGEMENT **1**
- *Introduction* *1*
- *Frontloading* *1*
- *Innovation* *5*
- *Visualization* *7*

PLM AND INTERNATIONAL PRODUCT DEVELOPMENT **11**
- *Results and reflections* *13*
- *Conclusions* *17*

LEAN SYSTEM DEVELOPMENT AND TESTING **19**
- *System development in the Automotive industry* *19*
- *Testing automotive systems* *19*
- *Lean System Development* *21*
- *Tests according to Lean and Agile* *23*
- *Testing in Japan* *24*
- *Discussion* *26*
- *More Lean than Agile* *26*
- *Testing to learn* *27*
- *References* *28*

PLATFORM-BASED DEVELOPMENT **31**
- *What is the general idea of a platform-based design?* *31*
- *Product Platforms* *31*
- *Production Platforms* *32*
- *Integrated Product and Production Platforms* *32*
- *Different Settings for Development* *32*
- *Cultural Differences and Their Impact on Platform Approaches* *34*

ROBUST DESIGN AND QUALITY ASSURANCE **37**
- *Introduction* *37*
- *Industrial visits in Sweden* *37*
- *Quality and Reliability Methods* *38*
- *Quality Supporting Activities* *41*
- *Summary* *43*

PRODUCT DEVELOPMENT MANAGEMENT

By: Ludvig Alfredson, Azadeh Fazl, Katarina Lund and Björn Söderberg

Product development management concerns how organizational issues (development processes, methods and tools) can lead to an increase in effectiveness and efficiency. In later years, the field has been affected by Japanese methods, mainly from car manufacturing companies such as Honda and Toyota. Under the label "lean product development," the field has gained much attention in both industry and the research community.

Introduction

We chose four research areas within product development management where an investigation into Japanese methods and processes was justified. First, we started with an article published in 1986, which showed that Japanese companies were superior in the way that they managed to front-load their workload in the early phases of development. We set out to investigate whether this was still the case in the first focus area, frontloading. Second, there has been research showing that Japan is one of the top countries in the world when it comes to innovation [1]. At the same time, there are concerns in Sweden that principles adopted from lean production might hamper the creativity and innovation climate in a company if implemented in the wrong manner. Therefore, our second focus area was within the field of innovation management. Third, many Swedish companies that are implementing principles and methods from Lean Product Development focus much of the early implementation efforts on visual management methods. Since Japan is regarded as the source to these methods, an investigation into how they use them in product development was justified. Finally, we chose the concept of the Chief Engineer as our fourth focus area. Described as having total responsibility for the development of a vehicle at Toyota and very unlike project and product managers in Swedish industrial companies, this role was interesting to investigate further. (With Toyota we mainly refer to Toyota Motor Corporation.)

Frontloading

Frontloading in product development simply means that it is in different ways advantageous to put emphasis on the beginning of a development project, to avoid costly changes and rework at the end. Japanese product development, in particular at Toyota, is quite well known for its ability to frontload. Frontloading is visualized in a graph in an article (shown below) from 1986 showing the difference between Japanese and American companies, where the Japanese are clearly better at frontloading. So, we set out to see if we could find indications that this still is true. After all, the study was done about 25 years ago. We were also interested in finding out more of what is behind the graph: what could be the explanation for the results?

The graph shows a comparison of Japanese and American product development with regards to the amount of engineering change orders over time in a project [1]. The difference is quite clear. Having lots of engineering change orders at the end of a project is devastating, because it is hard to estimate the time needed for execution and the spin-off effects of the change can become quite extensive. Moving the work-peak to an earlier point in the project reduces the risk in the project, and ultimately also the total workload. In general, companies struggle to achieve this. One reason why it is so difficult to start working more frontloaded is that companies seldom work on one project only. This leads to a prioritization of projects, and one of the most important selection criteria then is how much time they have before deadline. Those projects get the resources they need, which makes it difficult for other projects to frontload. After a while, those projects become the ones with an urgent lack of resources. This vicious cycle, combined with “the student syndrome”, i.e. that people tend to do the work as close to deadline as possible, makes it hard to frontload. This is confirmed by one of the Swedish companies. They stated during the interviews that they obviously focus on the late phases of the process, and the projects that are closing in on the delivery deadline, because “that’s where the money is.” They are used to working like that.

There are several ways of working more frontloaded, and we are interested in finding out by what means the Japanese product developers do this [2]. We got somewhat different responses when we asked about the graph, ranging from “this graph is very exaggerated,” to “It still applies, for sure.” In the following text, we will try to analyze the discussions we had regarding whether Japanese product development is more frontloaded than western, and if so, why? Three different topics emerged during the interviews in Japan: Late decision-making, Quality

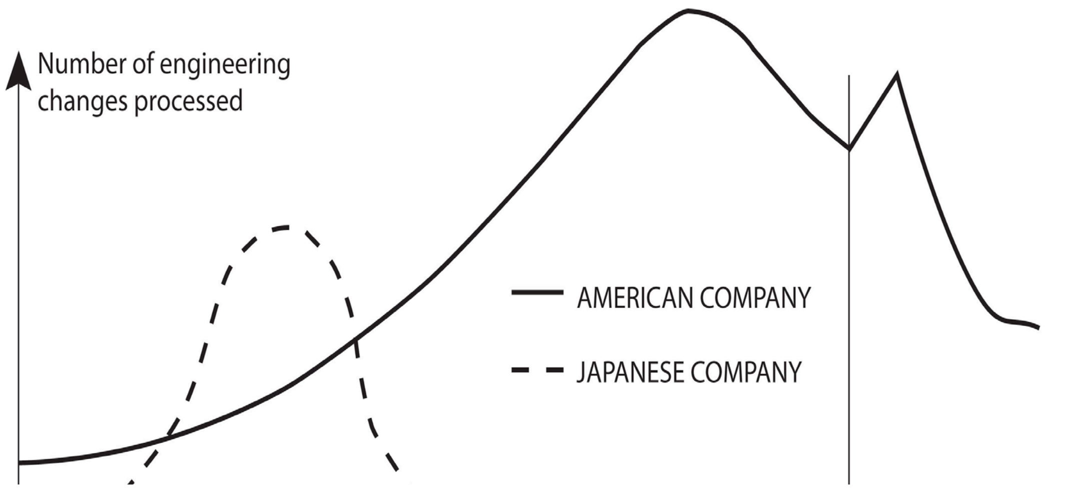


Figure 1: A comparison of front-loading behavior between American and Japanese companies (adopted from Sullivan 1986)

Function Deployment and Computer Aided Design.

Late decision-making

In our opinion, this is the most interesting explanation of the graph. One of the professors we talked to emphasized very much the ability of Japanese companies to make important decisions as late as possible, to avoid excess commitment. In other words, what do we have to decide now, and what can we decide later? Excess commitment leads to the unnecessary narrowing of scope, which almost inevitably renders more engineering change orders. If we make the decision late, we make it with a higher level of knowledge, which should make the decision more rational, and rational decisions mean less work. Also, the cost of changes in design increases logarithmically during the development process, which motivates making the decisions late, as that will render fewer changes. There is also less time for the decision to become obsolete, since it is closer to the deadline. One of the companies we visited pointed out excess commitment as a big difference between their development process and the process at a Swedish company. The Swedish company had put their product cost analysis much earlier in the process than the Japanese company (in other words, they committed too early to something that is subject to change). The Japanese development manager said in frustration: "How can they know so early, what the product cost will be?" The Japanese philosophy seems fairly simple: don't make commitments that you don't have enough data on.

One way to delay decision-making is through the concept of set-based engineering, also referred to as "the second Toyota paradox" [3]. The idea is to develop several alternatives at the same time to reduce the risk of selecting the wrong concept in the beginning, which will lead to a significant number of problems later. The notion of set-based design has received quite some interest in recent years. Nonetheless, examples of companies that have tried to implement it seem to be hard to find. We did not find much indication that a set-based approach is typically adopted in Japanese product development. One of the professors stated that the idea of set-based engineering comes from a supplier to Toyota. It cannot be used on a complete vehicle level, but rather on component level, where the development of prototypes is not too expensive. This also means that when the complexity of a product increases, the application of set-based engineering will decrease, since it is done on sub-system level. In other words, it

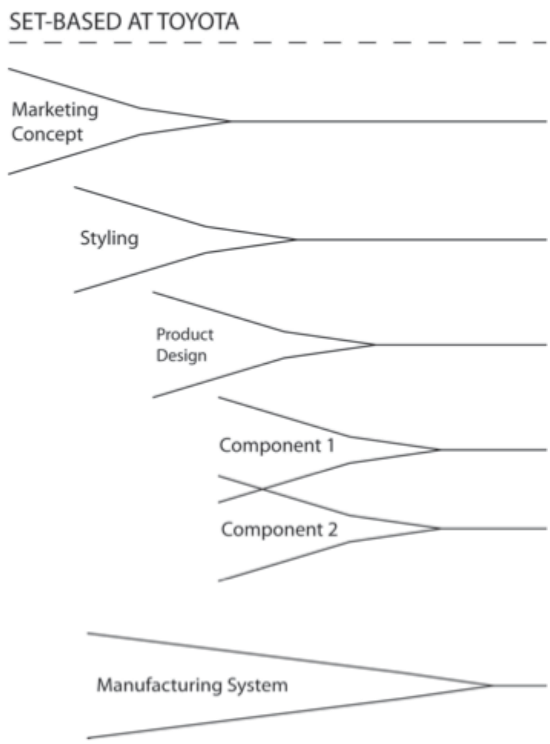


Figure 2: A description of Set-based engineering (adopted by Ward et al. (1995))

will probably be more difficult to develop components set-based if they have many interfaces to other components. One key to succeeding with set-based is close collaboration with suppliers, something Toyota is believed to be good at through their tight supplier network called the "Keiretsu-system." To conclude, it seems hard to determine whether a company is set-based in their development, because it is difficult to tell how far the parallel solutions are developed. Any company would say, "of course we look at different alternatives." The question is if they investigate them enough to make a rational decision to exclude non-feasible alternatives, or if, in fact, their gut feeling is the method they use.

Quality Function Deployment (QFD)

The use of QFD in Japan is how Sullivan [1] describes the graph in Figure 1. Japanese product development organizations are good at collecting and managing "the Voice of the customer". They do it systematically and thoroughly, and they use the QFD method. This goes hand-in-hand with the notion of "customer value," which is central in Lean production. The idea is that if we know up front what the customer wants, we have a better chance of steering the project in that direction. (We also know, however, that the desires and needs of the customer change continuously.) We interviewed a Japanese expert on QFD, and he actually uses this graph as a selling point for his consultancy on QFD. Interestingly enough, he was quite skeptical towards it, stating that there is a risk of overestimating the difference. That is because the article might be somewhat biased as it was written at a time when Japan strived to promote their stance as a country with good manufacturing capabilities, starting with the book, "*Japan as number 1*" [4], written in 1979.

Computer Aided Design and Simulation

A third topic that arose a great deal when we talked about frontloading is the use of CAD systems and simulation software. One of the companies talked about how much time they had saved through investing in new simulation software that could help them evaluate ideas much earlier. Two of the professors stressed the importance of having access to advanced simulation tools to evaluate the ideas in a swift and reliable way, and to get the input they need to be able to tell if a concept is good enough. However, the idea of frontloading through computer aid is not uncomplicated. Another professor pointed out the danger of letting CAD systems force the product developers to detail their design too early, which leads to the excess commitment discussed earlier. The computer needs the details to be able to simulate. If the details are not there, this will stress the organization into making irrational decisions. In whichever case, the Japanese are not famous for their CAD-systems. So even if it proves to be an important piece of the puzzle, it is unlikely it is the single most important success factor in frontloading.

Finally, it seems that Japanese companies use frontloading as a means to improve product development performance. A key seems to be the view of risk. Japanese culture is known to be risk avoidant. In this aspect, Swedish product developers can learn from Japan and acknowledge the fact that bringing uncertainty into a project with a tight budget and timeline often causes problems that are costly and jeopardizes quality.

Innovation

Innovation is recognized as one of the most important aspects of a modern economy. However, innovation is very hard to foresee and plan on. Rather, it is a result of a successful integration of creativity, efficiency, cost and functionality trade-offs and good marketing. Japanese companies have long been a role model for Swedish companies when it comes to production. Are they as good when it comes to the fuzzy front end and the creation of novel technological solutions?

Methods for creative activities

We set out to investigate if the Japanese tradition of using structured methods to support the way of working also occurs in more creative activities, as it does in, for example, production and testing. It turned out that many of the people we met were more of the opinion that creativity is in the human mind and has nothing to do with methods. One of the professors we met expressed it in the following way:

"Developing is head, not method! The important thing is how to find creative people."

The responsibility of finding these creative people lies with the line managers, but there are no assessment tests or the like available to help identify them. This selection is made from gut feeling and from what you, as a manager, see in people as they grow in their professional role. One company talked about the twice yearly occurring "career talk" where the line manager together with the employee decides in which direction he or she wants to develop. It is also in that meeting that employees with creative or leadership skills are encouraged to take on further responsibility in that direction.

Even though nobody explicitly mentioned specific methods for idea generation, there are many methods described in literature on Lean production [5] and Lean product development [6] that may help in creating more innovative products. The tradition of so-called "go see" applies not only to the workshop floor but also to getting to know your customer or in learning how products and systems work. The tradition of finding the root cause with, for instance, the help of "five whys" also helps to define the problems and make sure you address the right challenges.

We met a consultant coaching company working on the use of TRIZ, a method for idea generation and problem analysis that is rather widespread. It has, however, Russian origins and is not very spread in Japanese companies. He estimates that about 10% of Japanese engineers are familiar with the method, but, due to its complexity, far fewer actually use it. We did not hear of any comparable Japanese method.

The less described early phases of innovation

Japanese product development methods, especially those of Toyota, are well described in literature. There is, however much less written about the early phases of technology development. From what we heard from the companies we visited, there seem to be three consecutive processes: research and technology development that does not aim at a specific project, preparation of technology for a specific project and the development of a product concept to a complete solution ready for production. The processes and methods mentioned in literature mostly describe the third process.

In three of the four Japanese companies we heard of, the research organization was geographically separated from the development organization. One professor guessed that this was to keep the research organization less influenced by the demands of manufacturability and feasibility that the development organization desires. Only when the technology is mature enough does it move over to the development organization. Whether a technology is mature enough is decided based on a "sixth sense" judgment, as one company expressed it. When a technology is considered for introduction in a product development project, it is first tested thoroughly to make sure that the company knows enough to successfully integrate it with the rest of the product. One company talked about a typical eighteen months for such an intermediate process. As a technology or concept is moved from the research organization into the development organization, a few engineers normally move along with it. A great deal of documentation and drawings are handed over as well. It is noteworthy that the project is handed over to a new organization with another purpose – to make the product at a minimum possible cost and in the minimum time possible while maintaining quality.

There is a difference in how creativity manifests itself in the research organization and development organization respectively. Whereas the research organization strives to create new technology, the development organization is more oriented towards continuous improvements and manufacturability. As one senior engineer expressed it:

"Every engineer thinks every day, and every night, about how to improve their components. They know there will be problems and strive to find them."

One of the Swedish company representatives with some insight into Japanese product development compared it with a supermarket. The purpose of the research organization is to develop products that will end up on the shelf - new technology for the Development organization to "shop." To take this metaphor further could be to describe the development organization as the shop customer that picks up the ingredients at the supermarket for a dinner that he takes home, prepares, cooks and then serves to the guests.

Creativity as an individual capability

We will let one of the professors we met summarize many of the findings from our visits when it comes to creativity:

"There is no substitute for imagination!"

The Japanese way seems to be oriented towards the employee as an individual with a certain capability, and methods are not a very powerful aid in creative activities. If we instead shift to study *innovation*, methods may play a more significant part. Innovation entails not only coming up with something new – the activity we often call creativity – but also turning ideas into a product that suits the market need. Innovations are also what ultimately make a profit for the company. We saw that methods and traditions commonly described as Japanese probably aid innovation. A root cause analysis of different kinds helps solve the right problem and "Go see" and the "customer first philosophy" create a deep understanding of the problem and the end-user. Highly trained engineers with expert skills selected specifically for their creative abilities make sure good ideas are born, and the testing, control systems and tradition of continuous improvement ensure high quality.

There are, however, some Japanese cultural characteristics that could be obstacles to creativity, which is, in turn, considered a prerequisite for innovation. Without novel ideas, there will be no novel products. A creative climate is known as a work environment where things like dynamism and liveliness, debates and risk taking are advantages [7]. These are qualities that contrast the Japanese controlled culture, in which uncertainty, avoidance and conformity is strong. Nevertheless, breakthrough products are still being made in Japan. Is the novel technology a result of a few ingenious individuals supported by an organization specialized in turning their ideas into products ready for the market? After all, it only takes one person to come up with a good idea, but it takes a whole organization to turn it into a product available on the market.

Visualization

In the Swedish companies that we visited, there was a new method that dominated planning of project and line groups called "visual planning." It has gained increased popularity as a first method to implement when a company wants to adopt the principles described in Lean product development. It is a very simple method, using sticky notes to plan the development activities. It has also proven to be very powerful, creating benefits for the execution of the projects.

Background

In recent years, there has been an increased interest in the principles and methods of Lean product development, a management philosophy that has its roots in studies of the Toyota product development system. Many Swedish companies have started to implement this system, and the first step to do that has been through the use of visual planning, a simple visualization method for planning and communicating within and between projects.

Visual planning is a method where a group within the development organization uses an accessible board or wall for planning line activities or projects. In projects, this can be done in a dedicated room, often called Obeya [8] or War room. At these boards or rooms, the groups meet and update the information at short intervals, often several times per week. The planning is often done through visual artifacts (such as Post-it® notes) without support by IT systems, and is used for project planning, monitoring and execution. Some of the benefits that companies want to achieve using this method are increased communication, cross-functional integration and an increased commitment to deliverables.

Figure 3: Example of Visual Planning in a development organization

"You get a real update in the projects on a daily level. The project manager knows at nine o'clock exactly what is going on, all the deviations and all the risks. You know what is happening in the near future and have synchronization between all eight members of the projects. Through looking at a wall, you get very good communication within the group."

-Process manager at a Swedish company

Visual planning in Japan

Since both Lean authors and management consultants have described this as a Japanese method, we found it to be a very interesting research topic to investigate. Specifically, we looked at how this method is disseminated in Japanese companies. Questions like "Have these methods evolved further?" and "Do they use them in both a project and functional setting?" circulated in our heads and the question of whether Japanese companies implemented these methods never came up. The culture in Japan is very visual, and a good example of this is the widespread popularity of manga and anime, Japanese comic books and movies.

You can see old businessman sitting next to 12-year-olds reading the same manga magazine. This is very different from the western culture where you are expected to stop reading comics at a grown up age. This has been explained as one of the reasons why methods like visual planning have appeared in Japan.

At the companies, hope of finding visual planning was raised when the production facilities were shown. Many of the principles familiar from Lean production literature were there, such as 5S and competence matrices that clearly showed the workers' competencies. In the offices, there were many charts and plans printed and posted on the walls. This also showed that the production facilities were very visual in their information sharing. However, this was not the case when it came to the product development organizations.

After having some trouble understanding each other at both companies we had visited so far, we got the clear answer that they used neither visual planning methods nor dedicated project rooms in their product development organization. Furthermore, they did not use any project management software, such as Microsoft Project, that has become so common in western companies. Instead, the dominating software for project planning was Microsoft Excel. None of us in the research team associated this software with project planning. However, when interviewing a Swedish process manager with experience of working at Toyota MHG, it seemed as if Excel was used as universal software in Japanese companies. According to him, this software is used for planning, word processing, drawing processes and so on.

"At our company, we love Excel."

-Manager at one of the Japanese companies

When talking to the professors and consultants we had interviews with, all except one professor confirmed that they had never seen or heard of this kind of planning method before. The general opinion was that Japanese companies are moving more towards managing all information in IT systems, which is the trend that we have seen for many years in western companies as well. This was of course something that we had not expected, since visual planning methods are so strongly connected to Japan and Lean product development. However, the professor that had observed visual planning had seen it at some of the product development groups at Toyota Motor Corporation. This made us draw the conclusion that this method is something that is not widespread in Japan, but rather unique to some parts of Toyota. The perhaps somewhat ironic reflection here is that Swedish companies are implementing something they believe to be typically Japanese, which in fact appears not to be so. Yet research has shown that it is a powerful method for increasing communication, cross-functional integration and focus on deliverables. These benefits are more important to a product development organization than where they originated from.

Chief Engineer

You might have heard from someone that he or she is a Chief Engineer in his or her organization; however, in Product Development Management (PDM) literature, Chief Engineer (Shusa) refers to a concept originally derived from Toyota Product Development System. Many researchers have claimed that the success of Toyota Motor Corporation highly depends on its approach to product development; therefore, some western literature has described parts of the Toyota Product Development System [9]. Among others, the Chief Engineer concept has been characterized. Today, many Western companies, including Swedish companies, have also started to look into the topics in Toyota Product Development System in order to deploy them in their companies. As the concept of Chief Engineer is one of the key pillars in the Toyota Product Development System, it has also received a great deal of attention.

In the Toyota Product Development organization, a Chief Engineer is a person assigned to one product line (Corolla, for example) who owns a set of characteristics as well as responsibilities. At Toyota, a Chief Engineer leads all projects within his or her product line, following up all respective products in the market as well. As the Chief Engineer spends a lot of time with customers, in order to know them and understand their actual needs, he or she is a true customer representative in each project. The Chief Engineer is also in charge of market research and following competitors and market trends. For every single project, the Chief Engineer is responsible for securing the integration of different functions and roles into the project in order to achieve the targets. The Chief Engineer has also several assistants (future chief engineers) in the team. The number of assistants depends on the size of a project. At some Japanese companies, the numbers of assistants could be up to 50 people.

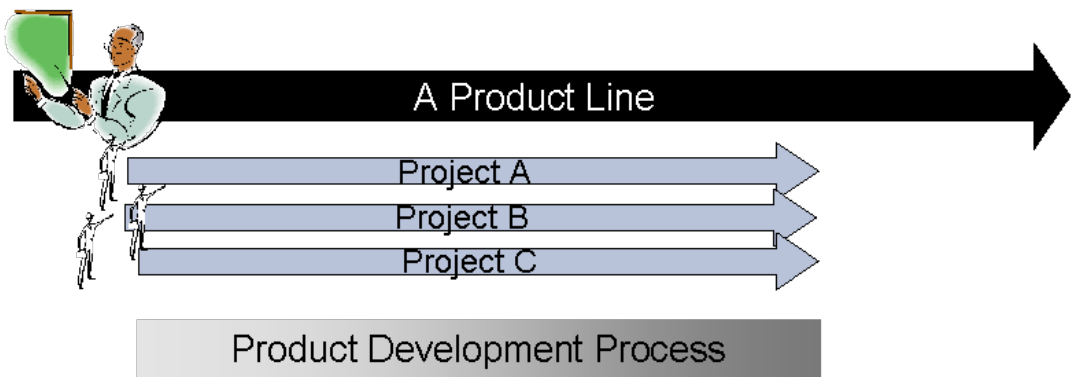
A Chief Engineer at Toyota is a knowledgeable, appointed person, having work experience from different functions for a longer period of time; therefore, the Chief Engineer's legitimacy is strengthened by seniority. A Chief Engineer is more than a project manager for a single project; rather, the Chief Engineer is a product line leader who has a holistic view of the history of products, the existing technologies, customer needs, ongoing projects, and future needed technologies and products. Because of all this knowledge, he or she is a well-suited person to create a concept vision for every new project in the line; the “guardian of concept” for each project. This role of Chief Engineer is an important part of the job. The Mazda MX-5 Chief Engineer, Mr. Hirai, described his concept vision as the synthesis of man and vehicle. He perceived this vision so strongly in his role that he also printed this description on his business card: "oneness between man and vehicle."

Beside this, a Chief Engineer has the responsibility of distributing knowledge to every project and transferring knowledge in between the projects. During project execution phases, the Chief Engineer is also involved in all critical decisions, and is aware of the exact project status at all times. The Chief Engineer steers the flow from "market to the end." A Chief Engineer also has a high capability for human interaction and communication. He or she often talks to many people, from sales people to other team members, and is dependent on human interaction.

Considering the above characteristics and responsibilities of a Chief Engineer, in application there are several ways to approach the Chief Engineer as a concept. During this study trip, we were interested to get better insight into this concept, to get further than simply reading western literature and their interpretation of the Toyota way, to understand how the Japanese way of PDM is done, and to suggest remedies for how it could support Swedish industry.

After visiting several companies, senior people and professors from both industry and university in Japan, we can conclude that the Toyota Product Development System is also unique in Japan and ahead of many others. This is especially true of the Chief Engineer concept. During our visits, we learned about different descriptions and approaches concerning the Chief Engineer from different professors. During the visits, we also got replies that were vague and sometimes hesitating; as a consequence, we had difficulties understanding the essence of the answers with respect to the concept of Chief Engineer. Generally, it seems that Chief Engineer as a concept is not established in many Japanese companies besides Toyota and some others. In the visited companies, we found one other company that had partly deployed this concept. From the professors we visited, we learned about a few other companies that have utilized the concept. One example is Mazda.

Among other job traits, the core of the Chief Engineer concept is his or her knowledge and first hand understanding of customer needs. It is also his or her power to influence the decisions in projects. By this, the direction and steering of projects is very customer-oriented, and the integration of customer needs in the projects does not have a great deal of handovers, which



create a risk of misunderstanding. Also, a smooth flow of information and effective communication is secured within any project due to the Chief Engineer characteristics as a leader.

Even though the Chief Engineer concept is not a common concept in all Japanese companies, it seems that many of them have succeeded in securing the core of customer orientation in their projects. In most of the companies we visited, understanding customer needs is the responsibility of everyone, from top management to every single engineer. Also, the middle and top managers involve themselves deeply in what is happening to customers and follow up on them continuously.

Reflecting on the observation reported above regarding the Chief Engineer concept, one suggestion for Swedish companies in this regard could be to embrace the above-mentioned core of customer-oriented project steering, as well as effective communication through the projects. In some Swedish companies, customer needs are not well integrated in their product development projects. In another words, sometimes the needs are not well understood. Furthermore, the ones who have the understanding of customer needs do not have the direct power to integrate them in the projects and influence the project directions. In some cases, there are also a significant number of handovers of customer needs in the process that make them vulnerable to misunderstandings. By adopting selected parts of the Chief Engineer concept, Swedish companies could perhaps be more successful integrating customer needs into their product.

References

1. Sullivan, L., *Quality Function Deployment.* Quality Progress, 1986 (June).
2. Thomke, S. and T. Fujimoto, *The effect of front-loading problem-solving on product development performance.* Journal of Product Innovation Management, 2000. **17**(2): p. 128-142.
3. Ward, A., et al., *The second Toyota paradox: How delaying decisions can make better cars faster.* Sloan Management Review, 1995. **Spring**: p. 43-61.
4. Vogel, E., *Japan as number one: Lessons for America*. 1979. Cambridge, MA: Harvard University Press
5. J. P. Womack, et al., The machine that changed the world : how Japan's secret weapon in the global auto wars will revolutionize Western industry, 1st HarperPerennial ed. New York, NY: HarperPerennial, 1991.
6. M. E. May, The elegant solution : Toyota's formula for mastering innovation. New York: Free Press, 2007.
7. G. Ekvall, "Organizational climate for creativity and innovation," European Journal of Work and Organizational Psychology, vol. 5, pp. 105 - 123, 1996.
8. Morgan, J. M. and J. K. Liker (2006). *The Toyota Product Development System: Integrating People, Process and Technology.*
9. A. C. Ward (2007). *Lean Product and Process Development.* The Lean Enterprise Institute, Inc.

PLM AND INTERNATIONAL PRODUCT DEVELOPMENT

By: Mattias Bokinge, Christoffer Levandowski and Anna Tidstam

Our group has focused on analyzing Product Lifecycle Management (PLM) strategies resulting from both cultural and company-specific situations in Swedish and Japanese industry. Our intention was to distinguish differences and similarities between the visited companies in Sweden and Japan. Continuous improvements of the system architecture and integral development are two major strengths of the Japanese PLM mindset. Globalization and the aging population are, on the other hand, two future challenges.

In this report, we define PLM as a business approach to creating and managing information about a product throughout its lifecycle by using a multitude of engineering tools, such as CAD systems and information systems (for example, Product Data Management (PDM) systems for product structure and data vaulting).
For studying the PLM situation during our visits to Japan, the following research questions were stated:

- RQ1: How does the use of PLM support in product development differ between Sweden and Japan?
- RQ2: What problems and future trends do the Japanese companies face when it comes to PLM?

With RQ1, we aim to pinpoint the differences and similarities in PLM support between Japanese and Swedish companies. For example, do they use in-house developed or commercial systems? How do Japanese companies develop their systems to efficiently support their product development process? With RQ2, we aim to achieve a deeper understanding of which problems Japanese companies are facing today. For example, what effects does globalization have on the PDM systems? And what are the future trends? The following chapter presents our results together with our reflections. This report then closes with a summarizing chapter, including our conclusions.

Results and reflections

The research questions are addressed in this section's chapters ”Continuous improvements of PLM systems,” “In-house developed PDM systems for product structure,” “Globalization and PLM strategy” and “Integrated versus modularized product development.”

Continuous improvements of PLM environment

Among other things, Japanese development is characterized by *Kaizen*: continuous improvement. *Kaizen* as a philosophy advocates small changes continuously, rather than making big radical changes. This set of mind can be found in engineering design, organizational development, process improvement and IT architecture development alike. Building on a firm ground of old knowledge to achieve higher grounds is the key to success. Japanese companies are good at “brushing up things through small improvements,” while Swedish companies are better at drawing up long-term strategies.

Errors do occur, even in Japan. The best way - the only way in the opinion of Japanese people we have talked to – is to learn from it and make it never happen again. As an error occurs, there are two things to do. First, assess the severity of the error, and then, depending on result of your assessment, apply a quick fix to the problem. The quick fix aims at mending the most acute damage caused by the error. Second, find out why the error occurred - a root cause analysis – and make it never happen again.

Japanese companies put great effort and create economic conditions in which it is possible into perform both steps, as stated by several of the people we talked to. Professors and company representatives have stressed the importance of *5 Whys*.

The *5 Whys* is a root cause analysis method that aims to find a chain of causes leading to the actual problem by asking yourself *why*.

The following is an example of the method:

1) My car will not start. (the problem)
2) Why? - The battery is dead. (first why)
3) Why? - The alternator is not functioning. (second why)
4) Why? - The alternator belt has broken. (third why)
5) Why? - The alternator belt was well beyond its useful service life and has never been replaced. (fourth why)
6) Why? - I have not been maintaining my car according to the recommended service schedule. (fifth why, a root cause)
7) Why? - Replacement parts are not available because of the extreme age of my vehicle.(sixth why, optional footnote)
8) I will start maintaining my car according to the recommended service schedule. (solution)

It is obvious that five whys may not be enough to find the real problem, which is why you should continue until you do so. As the chain emerges and the problem is located, you may direct your effort to the link you find solvable [2].

Japanese companies, as experienced as they are in this matter of mind, focus on finding the real root cause and aim the effort to finding a long-term solution (rather than just changing the battery). One company speaks of *saihatsu boshi*, which would translate to "make never happen again," as a way to develop their PLM environment. (According to Svensson et al. [3], their environment is composed of processes, several different systems, the information in the systems and the different roles in the organization). A root cause analysis is done every time an error (for example, in the design or in CAD Meta data) is discovered. Actions are then taken to make sure this error is never repeated. This action could be either a process change, a bullet in a checklist or a small implementation in the software.

The same company is remarkably well prepared for replacing one system in the PLM architecture, if viewed from a systems integration aspect. All integrations between the systems in the PLM architecture integrations are file based, which allows them to swiftly change systems without having to re-implement hard interfaces. However, it is our opinion that the small system implementations made for the sake of *saihatsu boshi* makes it harder (read expensive) to change systems.

- For one thing, the essence of every implementation needs to be extracted and implemented in the new system, which is time-consuming and costly.

- Second, the new system might not allow the desired customizations.
- Thirdly, after you have made all the changes to the system, it is not the same system anymore and perhaps not as effective as first estimated. Thus, the impact of the system implementations need to be excavated as a part of the pre-study of the system implementation project.

Last, customizing your system to the brink of recognition is possible. The risk is, of course, that the customizations have made the system to more resemble an in-house developed system.

In-house developed PDM systems for product structure

What is important to underline when discussing PLM strategies is that the companies we visited in Japan were all large companies, according the EU definition (more than 250 employees) [3]. What we have found during our previous research work is that many large companies in Swedish industry rely on in-house developed PDM systems for the product structure or for managing the documentation of the Bill of Materials. In the Swedish automotive industry, the PDM system for the product structure is, with very few exceptions, developed in-house. There are arguments that claim that an exchange of these backbone PDM systems to commercial systems would be beneficial. One reason, for example, is because of the high maintenance cost for in-house developed systems. Therefore, there have been attempts to customize commercial PDM systems such as ENOVIA and SAP Automotive to suit the automotive industry's needs [4, 5]. What we asked the Japanese companies was if similar scenarios are taking place in Japan, and if not, what are the reasons as to why they are not.

The Japanese companies and professors commenting on what is taking place in the Japanese industry did not have any information regarding any attempts to exchange their backbone PDM systems to commercial software. The justifications as to why this was not taking place were the following:

1) They do not want to let anybody else manage their data.
2) Commercial systems do not reflect the company's organization.
3) Their own systems are considered to be more user-friendly for their organization.

Our own reflections were that the Japanese culture of incremental improvements and high consciousness about getting all the details right does not go well with such a big change as exchanging PDM systems for product structure. The next chapter is about how Japanese industry is dealing with globalization effects, which has consequences with higher impacts on the PLM strategy than the previously discussed incremental improvements.

Globalization and PLM strategy

The companies we visited were in different situations: Japanese companies bought by Swedish companies, Japanese companies buying Swedish companies, Swedish companies starting up new subsidiaries in Japan, Japanese companies starting up new subsidiaries in Europe, and so on. What some of these companies are dealing with is the acquisition of already existing companies to obtain market shares and synergy effects. With the background of the justifications in the previous chapter of the in-house developed PDM systems for product structure, it does not come as a surprise that these acquisition companies are struggling with how to transfer to new PDM systems. One explanation for difficulties adapting to the new PDM system for product structure, taken from a Japanese company acquisition, was that communication with the production department was complicated. The strong collaboration between product development and production departments had not been supported by appropriate IT solutions with the new PDM system. A Japanese company acquiring smaller companies all over the world said that the reason why they have not yet succeeded with the global roll-out of their in-house developed PDM systems for product structure was that the translation to roman characters had not yet taken place. It was not obvious if that was the problem that caused the biggest challenges with the global roll-out. What we also found, which is quite evident but important to underline, was that the Swedish companies setting up new subsidiaries in Japan did not struggle with the globalization of their PDM system for product structure. Neither did the Japanese companies setting up new subsidiaries in the rest of the world. Our reflection is that this underlines the previous chapter’s motivation that the company’s organization is one of the most important reasons for why difficulties occur when trying to exchange the PDM system for product structure. The next chapter discusses another effect of globalization, the cultural aspect.

Developing Integral versus modular products

In products, different parts affect each other, and all parts jointly affect the behavior of the complete product. How much each part affects other parts and the complete product depends on the type of product and the development strategy chosen. A complete product may be divided into different modules. The idea is that the parts inside each module affect how the module behaves, but not how the complete product behaves. Instead, a set of modules together affect how the complete product behaves.

Different types of products are more or less easy to modularize. Computers are a typical example of a modular product. Each part, such as the motherboard and the CD-player, is optimized in isolation, with specified interfaces for integration into the other modules. In addition, each part can be re-engineered or replaced with only minor impact on the other modules. A car, on the other hand, is a good example of an integral product. The interfaces for the braking system modules, for example, are difficult to define. Each module therefore needs to be developed in close cooperation with the development of other braking system modules.

The strength of a modular product is that each module to some extent can be developed in isolation, and be replaced with other solutions. The weakness is that the interfaces constitute constraints on the freedom of development, making them difficult to optimize regarding, for example, space. The strength of an integral product is that it is easier to optimize, but the weakness is that it has to be developed in cooperation with the other parts of the complete product. Therefore, when developing an integral product, collaboration between developers of different parts is crucial.

Company representatives and professors in Japan emphasized that they are good at developing and working in teams. Prior to the visit to Japan, we had heard that Japanese culture was group-oriented, the US culture more oriented towards the individual and the Swedish culture somewhere in between. A Japanese professor stated that Japanese companies earlier produced many typical modular products, such as computers and displays. However, due to harsh competition, the development of this kind of consumer product moved to other countries. Instead, Japanese companies today produce typical integral industrial products and cars. Development in team is an important aspect in terms of communication. Our reflection is that people from Japan, in comparison to people from countries with a more individual-oriented culture, seem to be good at communication and that Japanese companies therefore develop more integral products than modular products.

In companies where product development is done in a global context, collaboration between individuals and groups is more difficult to pursue. This is due to different cultures, languages and time zones between different development sites, for example. Japan has traditionally had a big national market, and companies in Japan have thus been organized within the national borders. Sweden, on the other hand, has only had a small national market and has been dependent on international markets and collaborations with other companies outside of Sweden. Therefore, for a long time, Swedish companies have conducted product development in an international environment, as parts of larger multi-national organizations. Our reflection is that Swedish companies traditionally have faced bigger challenges regarding collaborative development than Japanese companies.

A professor stated that it is less popular in Japan today, compared to earlier, to send emails or use video conferences for communication. He said, although with a portion of humor, that a more common way to share knowledge in Japan was through drinking parties. We believe that most of the existing PLM support today focuses on special needs for different functions or on collaboration through information sharing. On the other hand, we believe that less attention has been put on collaboration support through direct communication.

Conclusions

By way of summarizing as well, our conclusions are that the Japanese companies visited during this study showed strength in their continuous improvements of their PDM systems for product structure, with the aim of preventing errors from happening again. In addition, we realized that the Japanese large companies also use in-house developed PDM systems for their product structure, as the large Swedish companies most often do. However, contrary to Sweden, the Japanese companies seemed to not even try to make any exchanges to commercial systems, and had many good arguments for keeping the in-house developed system. One of the arguments was that the company organization needed in-house developed PDM systems for product structure in order to secure that the PDM system efficiently supported the company.

Furthermore, we conclude that in an environment where the product development of integral products is conducted globally, collaboration in product development becomes more difficult to pursue. We believe that Swedish companies have an advantage compared to Japanese companies due to our longer experience regarding international collaboration in product development. However, the general focus within PLM has so far been on collaboration through information sharing. Finally, a future focus area might involve better support for communication.

References

[1] Wikipedia (2010). 5 Whys. Retrieved December 21, 2010, from http://en.wikipedia.org/wiki/5_Whys

[2] Schiemann, M (2008). “Enterprises by size class”, *Eurostat,* http://epp.eurostat.ec.europa.eu/ (Retrieved December 20, 2010).

[3] Svensson, D., Malmström, J., Pikosz, P., and Malmqvist, J., 1999, "A Framework for Modelling and Analysis of Engineering Information Systems," ASME Design Engineering Technical Conferences Las Vegas, Nevada, USA.

[4] Smidek, F (2001). “IBM och Dassault vill vara med överallt (In Swedish)”, *NyTeknik*, http://www.nyteknik.se/nyheter/it_telekom/allmant/article218056.ece (Retrieved January 24, 2011).

[5] CIMdata (2008). ”SAP supports Daimler in Global IT Harmonization”, http://www.cimdata.com/newsletter/2008/21/01/21.01.13.htm (Retrieved January 24, 2011).

LEAN SYSTEM DEVELOPMENT AND TESTING

By: Håkan Gustavsson, Mikael Adenmark and Daniel Sundmark

The Lean Product Development philosophy is derived from studies made of Japanese companies, especially Toyota. The overall purpose of this study was to gain a deeper understanding of how development is performed in Japan. A more specific purpose was to investigate how to make embedded system software testing more efficient based on the knowledge gained from the study.

System development in the Automotive industry

Software development within the automotive industry is not very different from that of other embedded systems. However, automotive embedded systems are characterized by being mechatronic systems, which adds complexity. The systems are often resource-constrained, and trade-offs between the system behavior and the resources required are of great importance. Cost, time-to-market and quality are the most important factors.

Today, most innovations made within the automotive domain are driven by electronics. According to a study made by Hoch et al. (2006), the total value of electronics in automobiles is expected to rise from the current 25% to 40% in 2010. Experts [7] estimate that 80 percent of all future automotive innovations will be driven by electronics. One of the reasons for the high cost of electronics is the large number of Electronic Control Units (ECUs) used. The trend in the car industry is currently changing, but there has been a philosophy of “one function – one ECU”.

Automotive customers demand new functionality with every new product release, and the time-to-market is constantly shortened. There are many new functions that are about to be introduced or have already been introduced that have a large impact on the electrical system of automotive vehicles. Further complexity is added by the fact that the vehicle developers strive to use a product line approach, where the same embedded system is used in a wide range of vehicles. The base system thus needs to be able to evolve over a long time and be adaptable to very different surroundings.

Testing automotive systems

A review of research in test, verification and validation in the automotive and vehicular domains reveals that contributions in this area are dominated by methods developed for the purpose of low-level model-based testing and verification. Studies focusing on integration- and system level testing of automotive systems are sparse, and this shortage is also highlighted by other researchers.

However, based on the studies that do exist, it is possible to derive a set of main challenges that has been reported regarding the system testing of automotive systems. First and foremost, there is reportedly a strong tradition in the automotive industry of building cars using the integration of modular third-party components. This is a tradition that has also been adopted by automotive software engineering. As stated by Pretschner et al, “In the past, the ’ideal’ of automotive development was that the parts of cars were produced by a chain of suppliers and more or less only assembled by the OEM. Thus, a large portion of the engineering and production activities were, and still are, outsourced.” [13]

While financially beneficial from a development cost per component-perspective (since standard automotive components can be developed once for several OEMs), one consequence of this sub-contracting culture is that the development process (including requirements engineering, implementation, integration and testing) needs to cross organizational borders. According to Grimm [7], having sub-contractors as part of the development process might significantly complicate system integration and testing (for example, in the form of communication being hampered by organizational and geographic distribution). Moreover, insights into the exponentially increasing corrective costs relative the time of defect detection in the development process in general software engineering has emphasized the importance of early testing, even at higher levels of integration. In the case of automotive software engineering, as in most embedded software development, system integration early on in the development process is often hindered by lack of target hardware access.

Another challenge, reported by Pretschner et al. [13], relates to variant handling. As a customer, you often want the opportunity to customize your vehicle. Different alternative selections of, for example, gearbox, engine or driver interface will also lead to corresponding selections in the electrical system configuration. This means that each individual sub-system selection needs to function properly in every possible integrated system configuration (or that the organization keeps track of which combinations that are incompatible and impossible to select). Naturally, the list of available sub-system configurations also varies over time. Further, with the long lifetime of automotive systems, backward compatibility needs to span over decades. Hence, when you test an automotive system, you test a family of systems, most often subject to combinatorial explosion.

Automotive systems are heterogeneous in nature. However, nearly all such systems include sub-systems with strict requirements on safety and reliability (e.g., braking and engine control). Testing plays a major part in the safety assurance of these sub-systems. This challenge is further complicated by the level of integration between the non-safety critical sub-systems and the safety critical ones. It is of utmost importance that the non-safety critical systems not be allowed to hinder the safe operation of the safety critical ones. In addition, this safety criticality also restricts the possibility of early testing in a real setting, since such testing would potentially endanger the safety.

Related to the above, there is a trend in the automotive domain towards more and more complex functionality, whose implementation is distributed over several previously isolated sub-systems. One example is electronic stability control, which, in the most complex case, requires interaction between the braking, engine control and transmission systems. Functions distributed over several sub-systems require performing (sub-)system integration before function testing. This, in turn, adds to the responsibility of integration testing.

Finally, Perez and Kaiser report on an observed overlap between different test levels in automotive systems development [11]. The authors state that “The strict

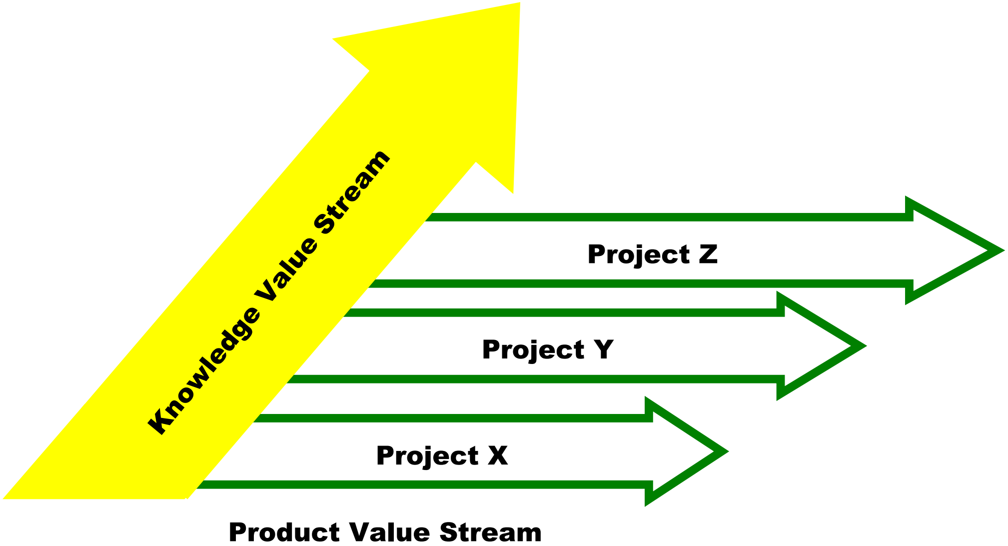


Figure 4. The two value streams of the product development [8]

separation of test levels results in similar or even identical test cases being separately specified, implemented, and executed at different test levels."

In summary, the main challenges in the high-level testing of automotive software, as reported by researchers in the area, are listed below:

1. To a large extent, automotive software is sub-contracted rather than developed in-house [7, 13] .
2. Automotive system verification is more difficult if sub-contractors are part of the development process [7].
3. Early integration testing is hampered by hardware dependencies and a lack of hardware availability [1].
4. The mass-customization of vehicles calls for massive configuration and variant testing [13].
5. The safety-critical nature of some vehicular sub-systems poses additional requirements on their verification [7, 13], including safety concerns of early system testing in a real setting.
6. A steadily increasing number of functions are distributed and require integration for testing [13].
7. Overlapping tests at different levels cause a waste of testing time and resources [11].

Lean System Development

The concept of Lean derives from the production methods developed by Toyota in the 1950s. The Lean philosophy has since then been applied to many diverse areas of operation. Software development has also been inspired by Lean, giving birth to SCRUM, Agile [15] and Lean Software Development [12]. It is important to note that none of the seminal work of Lean Product Development was made by people working inside Toyota or by native Japanese persons. Therefore, the available knowledge is more to be considered a western interpretation of the Toyota Way. Still, the success of the Lean philosophy is without any reasonable doubt.

During the study trip to Japan, we visited a number of companies and personally experienced the culture that gave birth to the Lean philosophy.

It is commonly known that the term Lean is not used in Japan. Nonetheless, many companies mentioned they had been inspired by the Toyota Way. Baines et al. [3] present the result of a systematic literature review of what is meant by the term Lean in product development. One finding is that the definition of Lean is drifting, moving from waste reduction towards value creation. Another result is although value is added in product development when useful information is produced, value needs to be defined precisely. Lean development focuses on creating re-useable knowledge - knowledge that contributes to the profitability of future operational value cycles and ideally can be used for many projects [16].

Within the Japanese culture, there is a very strong dedication to following rules and avoiding errors. Lean literature often mentions how work should be standardized in order to ensure quality. In our experience, it is easy to make a standard, but very hard to make everybody follow it. Visiting a Toyota factory, we noticed how everybody at the plant indicates with their hands that they look right and left before crossing a street. Finally, they pointed straight ahead before crossing. During the entire two-hour stay at the factory, we did not see anyone breaking this standardized way of crossing. Independent of the number of people crossing, everybody did it according to the standard. This practice is called pointing-checking (yubi-sashi-kakunin), and is a common Japanese practice to deal with safety checks. In development, other methods are used to ensure quality. The main tool we experienced during different company visits was the extensive use of checklists. Checklists are used during all different steps of the development processes. The checklists reflect the engineering knowledge accumulated over time.

Similar to our experience, Kennedy et al. argue that Toyota standardizes their knowledge into checklists and reviews all their design against these standards. Those checklists are updated after every project. Kennedy et al. argue that product development at Toyota consists of two value streams (Figure 4) [8]:

- The product value stream is unique to each project. Project X is not started until the alternative designs have been evaluated and decided upon. When the project starts, the risk should be very low. Knowledge acquired during and after the project is fed back into the knowledge value stream.
- The knowledge value stream consists of knowledge generalized for visual flow across projects and organizations. Checklists and A3 documentation are used to carry the knowledge.

Japanese companies invest very much in the training of new employees. New students are therefore educated during the first years at the company [10]. One company mentioned that 70-90% of the time was spent on education during the first year and 50% during the second year, another company mentioned that even math was taught.

One possible explanation for why companies invest so much in their employees is the Japanese concept of the lifetime employment. If engineers move, they usually move from OEM to supplier. It is rare to move from supplier to OEM.

Opposite to what is customary in the western world, salary generally decreases when Japanese employees change jobs [6].

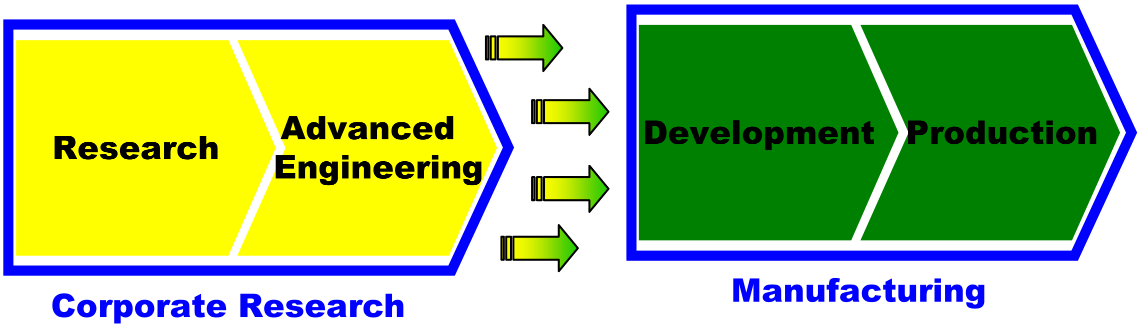


Figure 5. Organizationally and geographically separated functions.

Lean literature argues that companies should establish long-term relationships with a small number of suppliers, so you can really know them, and they rely on you for business [16]. Studies made by Fujimoto and Clark [4] show that Japanese OEMs involve suppliers to a much higher degree than American and European OEMs do.

The companies we visited seem to be working closer to the supplier than western companies. This is acknowledged by a study of patent applications [9] made by Japanese automotive OEMs and their suppliers. It shows that the ratio of shared patent applications made by Toyota is twice as high compared to its Japanese competitors. This study indicates that Toyota works very closely with its suppliers in the early phases of development. It also shows that Toyota in this regard is different from its Japanese competitors.

All visited companies had separated research and manufacturing, both organizationally and geographically (Figure 5**Error! Reference source not found.**). Corporate research seems to include what is commonly defined as research and advanced engineering. Manufacturing we visited included development and production. Development is located next to the production site in order to support production. During our visits, two explanations were given for why research and development is separated:

- They need to be close to the top universities (to attract top students).
- They need to keep the research disconnected from production and development.

Having separated organization for research and development could have complications on the architecture of the system. During our study trip, there was a large interest in reuse and product modularization among the companies we visited, although there was no hard evidence found that the reuse of components was effected.

Tests according to Lean and Agile

There is much literature on Lean production, but literature on Lean product development also exists. In the western software world, agile software development is also very commonly spoken about. Where agile is more handling methods, Lean is more of a philosophy directed towards a complete company. However, agile is derived from the Lean philosophy and adapted to software development. In the testing community, agile and Lean have grown strongly the last few years.

Agile testing methods, like scrum, focus on improving internal processes, in a fashion derived from the Lean principles. Some of the most important agile principles taken from “Agile Manifesto” [2] are:

- Frequent deliveries to and feedback from the customer allowing late changes in requirements
- Face-to-face communication
- Self-organizing teams
- Simplicity, maximizing the work not done (unneeded features)
- Frequent reflections to improve efficiency

The agile methods commonly involve testing as a part of the process [5]. Before the delivery of software (or, in scrum language, before the end of the sprint), the software should be tested. By testing continuously during the project, you avoid the bottlenecks at the end of the project as might be the case in traditional testing. By delivering to the customer frequently, you avoid developing something the customer does not want. This happens when requirements are interpreted differently by supplier and customer. You thereby avoid unnecessary work done (i.e., waste) for both customer and supplier. By testing simultaneously during development, you also avoid waterfall development, with all its handovers with built-in delays and loss of knowledge. Also, system testers, normally involved late in the development process, may be part of the early phases through early review and involvement in architecture design. Good system testers carry valuable knowledge of problems from similar projects and testability difficulties with some designs. They normally have a good overview of the final product. Testing needs more effort if the requirements are not perfectly clear and the delivered software is poor. However, it enables faster feedback to the developers, which shortens the combined time needed for development and testing.

Agile also emphasizes doing the most important work first. For testing, this means a prioritization of the tests by putting more effort into the high risk requirements first, where risk is normally evaluated as the product of probability and consequence of failure.

Lean focuses more on the complete chain from idea to end-customer. This is probably even more important to larger companies, such as automotive companies, where it is difficult for a small group of people to see the whole. This may, for example, involve making parts of the process less efficient in order to make the complete chain shorter. System testing would in many cases be affected positively by this, by requiring more testing done in earlier phases. However, on the other hand, it may also mean more work in some cases, as often system testing possesses expensive testing equipment that may be used for early development tests as well. Lean also handles longer term perspective than agile, which normally focuses on improving the short term process. If you do not invest time in long term changes or details sometimes (even those that are not presently that important), you risk increasing your technical debt to where it gets more and more difficult to handle short term problems in the long run as well.

Testing in Japan

”Too many checklists” was mentioned as a problem by a manager at a company we visited. However, it is very likely that the checklists are one of the big issues for why Japanese companies are considered both to be more efficient in their product development and to have higher quality.

Both developers and testers seem to follow checklists during product development. The checklists carry knowledge of problems from earlier similar work. This can be, for example, a guideline for “how many pressure cycles are needed in order to be certain enough on the quality of the test object.” A mandatory part of the development process is updating the checklist with new knowledge. If a field quality issue is found in a Japanese company, the corresponding checklist for development will be updated. In a western company, it is likely that the same field quality issue would result in a new test case in order to find the fault next time. Notice the difference in making sure the fault is never again created.

The Japanese mentality supports the sense for quality. Japanese people are very devoted to both process and the mind-set of not delivering any faults. As everyone expects all the others to have the same mind-set, it makes it easier to avoid double-testing (i.e., making very similar tests on more than one test level). An example of this is one automotive company where complete vehicle system testing contains no functional testing. The functional testing should already have been proven by having high quality in lower test levels. The same company sometimes had its supplier performing the integration of its components. Generally, Japanese companies seem to work closer to their suppliers, and this is in regard to testing activities, too. Another Japanese company mentioned that it often very late (the day before) got knowledge about new SW/HW being delivered for system testing. This was possible since the integration had then already been done before delivery, which makes unplanned problems during testing less likely. A common statement in the testing community is that the cost of finding a fault becomes 10 times higher for each increase in test level. This theory is supported by the seemingly Japanese way of testing, where more test effort is spent on lower test levels.

However, it should be noted that the Japanese companies visited all worked relatively project-oriented with relatively integral architecture. With more variants, it becomes more difficult to test system software functionality on lower levels due to the many dependencies on other systems. It becomes more important to use the complete system for integration and regression testing. Several of the automotive companies we visited wanted their architecture to become more modular.

Another interesting issue identified at some Japanese testing departments is the lack of prioritization of testing activities. It seems obvious that you test everything that has been defined as an issue for testing. It is well-known in the testing community that it is impossible to test everything. Therefore, a prioritization of testing activities becomes necessary. However, if you can assume that you are working with a high quality system where problems are always and have always been highlighted directly and solved directly during development or even during the research phase, maybe the need for system testing activities become less important. With very little technical debt, maybe it becomes justified to only perform some basic functions that cannot be tested at lower test levels. It is not uncommon that faults found should normally have been found at lower test levels. If a Japanese tester finds a new fault, he or she makes sure the corresponding checklist is updated in order to make sure the problems do not arise again.

Development in small steps (i.e., avoiding changing too many systems at the same time) also makes it easier to identify dependencies and, thereby, also easier to test. Unchanged functionality already working in production needs less attention and testing. “No changed development is the best” was mentioned by a manager, and by that he meant risk minimizing.

A professor we visited who had good insight into Japanese product development mentioned that software development and testing is more end-loaded than other development. This is possibly due to the fact that most higher managers have a history in mechanical development. Therefore, it is still common that the software developers and even more so software testers are the ones working late nights well into the projects.

Discussion

In Japanese companies, it seems that more work is done during the research phase in order to be able to perform fast product development with the intention of mass production. Even the integration and assuring of a high quality of the containing components seems to be done in some cases before the start of mass production development. In harsh words, the product development could even be seen as an assembly line for product specifications and descriptions, a phase where little creativity is required, but the work can be done very efficiently. Rather, the creativity needed in product development is used for the continuous improvements to the re-occurring work and processes. A disadvantage of performing research separately and continuously making the mass production development more efficient may be that it makes modularization more difficult. With highly modular architecture, you need to know more about the end product, as you have more dependencies on other systems that are also changing over time. In other words, you probably need more contact with other developers in order to understand which kind of vehicle your component will end up in. Therefore, it becomes more difficult to integrate your system as early on as the research phase, and more effort needs to be put into the product development phase. Modularity also drives the need for time consuming variant and configuration testing according to challenge 4 above. However, we could not identify how testing and integration was performed during the research phases. How are components integrated before you know what the end product will look like? The only observation made was that new components are tested in old vehicles. Probably this is managed by expecting small changes in the product from project to project. In that case, it becomes more reliable to perform the integration of new components in old vehicles and in that way minimize the risk during mass production development. The component developers at the same time gain better knowledge of the end product. This is probably not that uncommon in Swedish companies either. However, the degree of assurance that the integration will work in the end product as well is greater if you need to consider fewer changes in other components.

More Lean than Agile

No (western) agile working methods could be identified. Some companies showed examples of how work planning was performed through visual boards, making sure each engineer was working only with one task at a time. This is the idea behind kanban, which is considered part of the agile ideas. Scrum, for example, was not known. You also find tracks of agile in the fact that the developers do most of the testing themselves and the teamwork approach to problems. The degree of early tester involvement is also high in Japanese companies. That is identified as one of the key areas of how to improve test processes in agile environments according to TPI Next [14]. However, although Japanese internal product development methods are not revolutionary, their sense for quality, details and immediate highlighting of problems makes the flow from development process to the end customer more efficient. This is due to fewer handovers, fewer double test efforts, internal fault handling and claims. Japanese companies seem to be more Lean than Agile, which may not be that surprising. However, from what has been identified at the Japanese companies visited, Lean is not necessarily equivalent to the Japanese way of working. For example, the approach of "testing everything," as was discussed in the previous chapter, does not comply to Lean, where prioritization is necessary. A prioritization says that you should test some parts more thoroughly. This in turn means that some parts will be tested less. That means you will be more prone towards accepting faults, and thereby increase your technical debt. Attending solely to the biggest problems may in time risk resulting in the smaller problems increasing in number and magnitude, resulting in a larger total amount of problems. Instead, a Japanese approach would probably be decreasing the number of changes in order to have enough time to develop and test the changes introduced. More focus is instead put on improving the developers' checklists in order to avoid faults being developed. The Japanese "testing everything" means testing everything that has been identified as necessary in previous similar projects. In other words, it does not mean that you should do exhaustive testing.

Testing to learn

As mentioned earlier, the Japanese use testing in order to improve development by quickly highlighting the problems and performing thorough root cause analysis. If each test level confirms the earlier test levels, you could cut down on a great amount of testing. As long as you double-test different test levels, you have a cut-back in test efficiency as mentioned in Challenge 7 above. It is not uncommon that testing is performed more thoroughly on systems where, due to experience, you do not trust the quality of the delivery. The possibility to learn gets lost through this way of working, a typical waste where no value is added in the testing other than reporting faults. This supports the earlier statement that having sub-contracted systems makes verification more difficult (Challenge 2), as you by definition then have handovers and normally perform some kind of double testing. Japanese companies, however, seem to work closer to their developers, thereby minimizing these wastes.

Checklists were identified as one of the major ways of keeping knowledge within the company, as described earlier. However, too specific checklists may lead to only tending to the points in the checklist and not thinking about the bigger picture. Therefore, it is important that checklists be more guiding than directing, at least as long as creativity could be useful, which is normally the case. Apart from the use of checklists, there is also a cultural difference here: Japanese people tend to stay longer within the same company and even the same department. This is more difficult for a western company to adopt. One of the big wastes described in the Lean literature is the loss of knowledge. With no handovers to new employees, there will be no loss of knowledge, at least during that time. New employees also seem to have much more time for education and on-the-job training in Japanese companies. This means less loss of knowledge as the companies grow or people do change positions. As Japanese people seem more devoted to the process, performing their work and fulfilling their responsibilities, they seem to get the job done no matter what. They get the bigger picture by highlighting the problems and performing root cause analysis. The root cause analysis is the creative part of this work. Western engineers are more devoted to automation according to the boring index, making the work they need to do often more efficient. There is nothing wrong with that, but if it is prioritized over finding the root cause of the problems, you may end up simply performing unnecessary work faster.

References

[1] A. Michailidis, U. Spieth, T. Ringler, B. Hedenetz, and S. Kowalewski, "Test Front Loading in Early Stages of Automotive Software Development Based on AUTOSAR," 2010.

[2] A. Alliance, "Agile Manifesto," *Online at http://www.agilemanifesto.org*.

[3] T. Baines, H. Lightfoot, G. Williams, and R. Greenough, "State-of-the-art in lean design engineering: a literature review on white collar lean," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* vol. 220, pp. 1539-1547, 2006.

[4] K. Clark and T. Fujimoto, *Product development performance: Strategy, organization, and management in the world auto industry*: Cambridge, MA: Harvard Business Press, 1991.

[5] L. Crispin and J. Gregory, *Agile testing: a practical guide for testers and agile teams*. Addison-Wesley Professional, 2009.

[6] T. Fujimoto, *The evolution of a manufacturing system at Toyota*. city: Oxford University Press, USA, 1999.

[7] K. Grimm, "Software Technology in an Automotive Company - Major Challenges," in *International Conference on Software Engineering*, 2003, pp. 498--503.

[8] M. Kennedy, K. Harmon, and E. Minnock, *Ready, Set, Dominate: Implement Toyota's Set-Based Learning for Developing Products and Nobody Can Catch You*. Oaklea Press , 2008.

[9] Y. Konno, "Enhancement of the advanced R&D cooperation between automakers and suppliers in the Japanese automobile industry," *Annals of Business Administrative Science,* vol. 6, pp. 15-34, 2008.

[10] J. M. Morgan and J. K. Liker, *The Toyota product development system : integrating people, process, and technology*. New York: Productivity Press, 2006.

[11] A. Pérez and S. Kaiser, "Top-Down Reuse for Multi-level Testing," 2010, pp. 150-159.

[12] M. Poppendieck and T. Poppendieck, *Implementing lean software development : from concept to cash*. Upper Saddle River, N.J.: Addison-Wesley, 2007.

[13] A. Pretschner, M. Broy, I. H. Kruger, and T. Stauner, "Software Engineering for Automotive Systems: A Roadmap,"

International Conference on Software Engineering, pp. 55-71, 2007.
[14] Sogeti, "TPI Next – Business Driven Test Process Improvement,": UTN Publishers, 2009.
[15] J. Sutherland and K. Schwaber, "The Scrum Papers: Nuts, Bolts, and Origins of an Agile Process," 2008.
[16] A. Ward, *The Lean Development Skills Book*. Ann Arbor, Michigan: Ward Synthesis, 2002.

PLATFORM-BASED DEVELOPMENT

By: Marcel T Michaelis and Stellan Gedell

One of the main drivers for platform-based development and production is the possibility to combine customization with economies of scale. The reuse of common resources in multiple, customized design variants is one way of achieving this. However, the economic benefits may be easily lost. A relatively small design differentiation, for example, may lead to a relatively large number of parts having to be modified. More fruitful approaches that can support carry-over without major drawbacks include, for example, adopting a carefully planned development process or reusing more abstract design knowledge.

What is the general idea of a *platform-based design*?

The concept of *platform-based development* is an approach to the challenge of designing profitable products. The basic idea is to strategically define what a company wants to reuse for a range of products that they produce. Numerous industries and companies adopt platform strategies for developing and managing product ranges. The underlying goal is to exploit benefits of scale while achieving distinctiveness and commonality across products [1]. Another goal is to plan ahead in time to accommodate technological development or changes on the market. Moreover, the notion of a platform can be used as an approach to manage and to maintain knowledge in the company.

For engineers, utilizing platform strategies can mean the reduction in the reworking of already working systems. In turn, this can provide them with more time for unique and value-adding development tasks. It can even help with structuring development projects and facilitating creating new products based on existing ones.

Platform strategies can be very unique for each company. In fact, during our visits in Sweden and Japan, we observed that the idea of the strategic reuse of assets across the company's products (a platform strategy) can vary significantly between companies.

The company's industrial context, its market situation, and organizational structure are only some few examples of what has an impact on the platform and makes it unique for a specific company. Platform strategies are commonly adopted, for example, in the automotive industry. Here, they originally mean the defining of certain product parts (e.g., a car's underbody) to develop different models based upon them. However, a platform can be defined, not only for the products but also for the production efforts of a company. Here, for example, the parts of a factory and the machines available for production, used for a range of products, can be seen as the platform.

Product Platforms

When it comes to product platforms generally, two prevailing extremes can be identified [2]. First, a physical perspective can be taken. In this perspective, the platform is represented by parts or assemblies the product is composed of. Second, a product platform can be seen more abstractly "as the collection of assets that are shared by a set of products," including components, processes, knowledge, as well as people and relationships [1]. There is a great deal of room for variation and company specific definition in between those two extremes.

Production Platforms

One can use a strategic approach to the reuse of assets in production as well [2]. In fact, because one typically reuses machines, factory halls and processes, for example, this comes rather natural. When one tries to plan production layouts and concepts, a platform might mean several things. For example, the company might want to ensure that a range of products can be produced with the same equipment. It could also mean that, thanks to strategically planning ahead, the production layout and concept can be used in a similar way over a long period of time, even though the products of the company change.

Integrated Product and Production Platforms

When the definition of a platform is very broad, it includes several aspects in one integrated approach. This can mean that the platform not only includes the product but even the production system and its related processes. For example, product and the production system can be regarded as two systems that interact with each other. Then "a set of subsystems and interfaces developed to form a common structure" [3] can be defined and products and production systems can be jointly developed with a pervasive strategy.

Different Settings for Development

The development of products (and production systems) happens in many different settings. Below, some of the factors defining different settings are presented. Their relevance for development in general and the utilization of platforms in particular are elaborated:

Nature of the product

The term product is a wide one, and many aspects can have an effect on a platform strategy. Prominent factors for the notion of platform are the complexity of the technical solution, the technical domains that are addressed (mechanics, electronics, software engineering, etc.), the question of how modular the products are, and how much integration work is needed to join all the bits and pieces into one product solution. With few exceptions, the companies we visited are from the manufacturing industry, with rather complex product solutions involving different domains.

Nature of development project

Naturally, products and its constituents can have varying degrees of novelty (for example, new development, adaptation, or configuration). A platform strategy changes development work in this respect by defining what to keep static over a certain period of time and what is subject to change. Some of the Swedish companies we visited in particular use platforms as a tool to strategically develop their product range over time. Through this, they also define which parts of their assets are subject to more rapid change within a given timeframe, and which are to be kept stable.

Volume of production

The volume of production has a direct impact on how economies of scale can be achieved. Here, we observed a wide range of different setups, from only a denary of units produced per day to several hundred.

Commonality across products

How similar one product is to the next in a portfolio has a strong effect on how platforms can be defined. Like the production volume, commonality affects how economies of scale are achieved. Companies we visited differed largely in this aspect. Some companies customize their products on demand with extensive accommodation of individual customers' wishes. Meanwhile, others had little variance within one model, but instead a large number of models by default.

Size of the company

The number of employees in general and the number of people involved in developing and maintaining products in particular contributes to setting the scene. Information and knowledge management is affected according to the number of people involved. All the companies visited had at least several thousand employees.

Ownership and organizational setup

If the company is part of a larger group, the strategy for reusing assets is set into a larger context. Standards might be imposed by the group organization and the daughter company must follow this standard when developing products. For example, platform strategies are common across products of individual brand organizations in the automotive industry to exploit economies of scale. This is also the case for a truck manufacturer we visited.

Market situation of the company

If a manufacturer targets different markets, geographically and with respect to the industry or branch, a strategic approach to diversifying products might be needed. The goal here is products with a unique design for each market. The companies we visited mainly remain within one line of industry (with few exceptions). At the same time, all companies have an international profile with domestic and oversea markets. We visited some companies that mainly supply parts and components, but the majority were OEMs.

Organizational structure of the company

The arrangement of departments and their affiliation with certain parts of the product, the production system and technology affects strategic planning and reuse. The approach of a platform has to take into consideration this arrangement to facilitate cross-disciplinary and cross-functional development work.

Geography

The distance between sites of research, development and production affects how companies arrange their work and how they implement standards. A platform strategy is a puzzle piece in this larger picture. The choice of suppliers, for example, might be affected by where sites of production are. All the companies we visited worked across national borders when developing and producing products. With respect to knowledge transfer and management, some companies we visited saw a challenge in aligning their efforts to effectively develop and produce products.

Cultural differences

Cultural differences also affect the manifestation of platform strategies. The following section reflects upon some of the cultural differences we observed and their possible impact on platform approaches.

Cultural Differences and Their Impact on Platform Approaches

If we are to believe literature and reflect upon what we have observed ourselves, we can hypothesize that Japanese are less inclined to take risks. One of the underlying drivers for this is to avoid *losing face*. Japanese will consider the potential negative risks connected with new designs more thoroughly. If still in doubt after careful research, a newer and more radical design idea might not be adopted. This might come at the cost that the advantages of a new design cannot be exploited. Consequently, Japanese might be more inclined to re-use resources, ideas, and maybe even physical parts, which often is the meaning of the term platform. Our findings during company visits point to this. Not too unlikely, drivers for re-use exist in western countries, too. Here, the reasons are rather economic ones. Exploiting economies of scale is a common theme. However, an opposing force can be identified, namely the individualism of the designers. It is considered more challenging to create new designs than to re-use existing ones. This is supported by western reward thinking, where it is important to shed positive light on one's own person, rather than one's group. This is somewhat in contrast to the Japanese mentality. There, uniformity and group thinking have a stronger stand. Illustrating this is the Japanese expression "Nails sticking out are beaten down."

Cultural aspects affect platform thinking in even other ways. The benefits of adopting a platform approach to designing increases with the number of derivate products based on them. The successful planning of the platform content and the coming derivate designs, along with the ability to follow through those plans, are vital. Here, Japanese aversion to taking risk is beneficial. Japanese designers will plan future activities rigorously and, as far as we could conclude, follow through with the plans. A possible drawback of strict planning can be a reduced capability to react to market changes. However, some of the companies had planned for such events, too. For example, a company regularly visited their competitors. The engineers studied the competitor's latest changes and quickly adjusted their customer offer accordingly.

Even in a wider sense, effects from striving to avoid risks can be seen in products and product development. These effects address the social setup in general. Communication within a group of Japanese is a way to ensure not becoming a maverick, someone who pushes through his or her own ideas and agendas. Communication within the team, team work and consensus in decision making are all positively affected in particular. Where we, as representatives for the Western society, were looking for IT-systems to support communication, there was simply less of a need for that in Japanese companies.

However, there are limits to how far interpersonal and group communication can reach when working on large projects with many stakeholders in many geographical places. Some designers expressed concerns that the increasing globalization of product development and production might make the usual communication tools insufficient. This has bearing on the development of platforms because, here, integrating systems and taking into consideration many different requirements are keys.

Of course, solving communication relying mostly on interpersonal communication has advantages and disadvantages. IT-tools that would be an alternative are narrow-banded and might be expensive compared to the benefits they provide. Generally, products with a lot of dependencies, often referred to as integrated products, will gain from well-functioning communication. Looking at the product and the production system, even more need for integration can be identified, namely between product design and production design. We were informed several times during our visit of the importance of having the plant close to the development department. (Note that Japanese companies' plants not located close to the development department are replicates of the mother plant.)

However, when products become too large, the informal communication starts to become a limitation. During our visit, we heard comments like this one:

"Japanese companies are good at integrated products with coupled design solutions, but they have not succeeded with really complex products like aircrafts and telecommunication systems."

With our limited number of studied companies, we would not be able to draw that conclusion. However, it seems reasonable that communication based on human contacts has a limited scalability.

Noticeable during the whole trip is the Japanese striving to refine to a higher extent than what we are used to. It is common in Western societies to talk about an 80/20 rule. That means to focus on the most important issues and, consequently, to care less about the rest. Even here, Japanese behavior can be deduced from the unwillingness to take risks. By leaving some uncertainty, there is of course a potential risk of problems emerging.

Being in Japan, we got a good general impression of the importance of planning and an affection for numbers and measuring in everyday situations. For instance, trains follow their timetables to the minute and signs indicate on which side to walk in the metro. Beyond this, we have also seen examples from companies we visited. We interpreted statements like "get facts from experiments" and "don't just get a feel" as coming from a desire to get quantified data that allows careful planning. Searching for a root cause, the question "Why" was an important one. One company, for example, showed drawings that contained more than geometrical information. There were multiple arrows and text blocks, probably explaining some detail. However, due to (intentionally) bad resolution on the presented picture and language difficulties, this is only our interpretation.

References

[1] Robertson, D. and K. Ulrich, Planning product platforms. Sloan Management Review, 1998. 39(4): p. 19-31.

[2] Michaelis, M.T., A. Lindquist Wahl, and H. Johannesson, Integrating Product and Manufacturing System Platforms - Exploring a Configurable System Approach, in 11th International Design Conference DESIGN, 2010, Dubrovnik, Croatia. p. 1605-1614.

[3] Meyer, M.H. and A.P. Lehnerd, The power of product platforms: building value and cost leadership. New York: Free Press, 1997.

ROBUST DESIGN AND QUALITY ASSURANCE

Robust design aims at reducing the effects of variation on a design without eliminating the variation itself. It is a method of improving product quality with little or no additional manufacturing costs.

By: Anders Forslund, Karin Forslund, Timo Kero and Ola Wagersten

Introduction

During company visits and expert interviews in Sweden and Japan, a study was performed. It investigated the differences between the companies when it came to product development methods. The next chapter focuses specifically on the area of Robust design and quality assurance. The following questions have been posed:

How can robust design practice in Japan and Sweden be described?

In posing this question, we wanted to identify differences and similarities within robust design practice in Sweden and Japan. Robust design practice is to be studied with respect to product design. It comprises activities aimed at minimizing the effects of variation without eliminating the sources of variation.

Can any differences in product development procedures influencing the implementation of robust design methods be found?

Another relevant question is whether Japanese companies explicitly refer to robust design as a methodology or whether practices and procedures associated with robust design are inherent in Japanese company culture.

Industrial visits in Sweden

In Sweden, two different company visits were carried out. The first visit was conducted at a supplier to the car industry. The company sets high focus on delivering products of high quality. Therefore, product development methods with safety focus were prioritized. The company was also very committed from a robust design perspective, since bad geometrical quality could negatively impact the reliability of their products. At this company, we also got to meet a group of experts and the group manager for basic development in Europe. A procedure that linked various robust design methods was presented.

The procedure had been implemented when some of the OEM companies had demanded that a package of methods was to be delivered from all their suppliers to ensure product quality. These OEMs were American companies, and the initiative to implement robust design methodology came from America.

At the other Swedish company, we met one Quality Coordinator and one Best Practice engineer. The company was influenced by the fact that they had been bought by a Japanese company. For their products, this had implied a shift from prioritizing premium products to a focus on manufacturing quality. As for the product development process, the Japanese company had had three main requirements on the company: implement Q-gates

(Quality-gates), develop a best practice system and limit trade-off curves.

At the first Japanese company, we met with two product managers and one project manager to discuss quality and robust design. This was our first meeting with Japanese engineers. Having recently been bought by a Swedish company, the company was in the process of assimilating their work practices with a development plan according to the new owner. This was interesting for us, since this could help us in our aim to find similarities and differences between Japanese and Swedish product development processes.

At the other Japanese company, they relied heavily on the usage of CAx tools in early phases of product development. This company also subscribed to the philosophy of *Kaizen*. In addition, this company promoted quality efforts by giving quality awards to employees who found new ways of improving quality.

The first Japanese expert had extensive knowledge and experience in the area of product development management, after 30 years of working at numerous Japanese companies. His focus now has turned towards questions regarding environmental issues and designing environmental friendly products. His specific method descriptions are included in the method chapter.

The second expert had more than forty years of experience from the field of engineering. He conducts his research today in Digital Engineering and Inverse Manufacturing (Environmentally-friendly Manufacturing). The third expert had been working at a well-known electronics company for 32 years. Today he works as a consultant for a number of companies. Two of his specialties are TRIZ and QFD.

The quality methods discussed at the companies will be described more thoroughly in the following sections.

Quality and Reliability Methods

Robust design aims at reducing the effects of variation on a design without eliminating the variation itself. It is a method of improving product quality with little or no additional manufacturing costs. Robust design methodology was pioneered by Japanese statistician Genichi Taguchi in the 1960s. As an engineer, Taguchi had gained great experience from several different Japanese companies and later in his career from American companies. Today, robust design methodology is employed in companies all over the world. Robust design focuses on improving the fundamental function of the product or process, facilitating flexible designs and concurrent engineering. It is a powerful method to reduce product cost, improve quality, and simultaneously reduce development gap. Ironically, the robust design methodology was only explicitly used at one of the interviewed Swedish companies, and at none of the Japanese companies interviewed.

One part of the Taguchi method, P-diagrams (Parameter diagrams) [1] (Figure 1), plays a large part in the quality assurance process at one of the Swedish companies. There, the diagrams were prepared for different product functions and properties. Influential noise factors and control factors were found. This was used to identify further failure modes. Statistical probabilities for functional deviations were calculated. Boundary limit values of parameters were at times analyzed using a design of experiment procedure. The company also compiled robustness checklists.

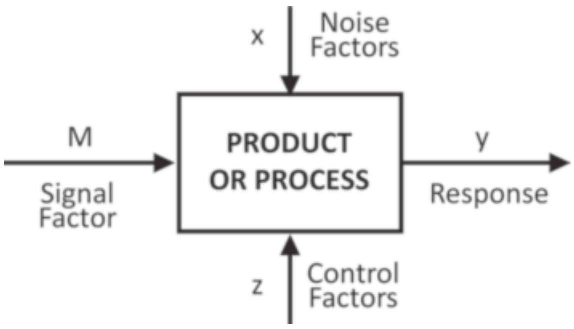


Figure 1. The P-diagram [1]

In Japan, one of the professors mentioned another related method - Statistical Quality Control (SQC). It is the application of statistical methods to the monitoring and checking of a process to ensure that it operates at its full potential to produce conforming products. Ironically, this method was developed outside Japan by Walter A Shewhart of the Bell Telephone Laboratories in the USA. The method is divided into three broad theories: descriptive statistics, statistical process control and acceptance sampling. The tool in each category provides different types of information for analyzing quality. SQC is a system of routine activities used to measure and verify the quality of the inventory as it is being developed. The SQC System is designed to:

(i) Provide routine and consistent checks to ensure data integrity, correctness, and completeness;
(ii) Identify and address errors and omissions;
(iii) Document and archive inventory material and record all SQC activities.

SQC activities also include general methods, such as accuracy checks on data acquisition and calculations and the use of approved standardized procedures for calculating emissions, performing measurements, estimating uncertainties, archiving information and reporting.

Quality assurance is sometimes set in relation to quality control. Quality assurance means making sure early on that quality is built into the product, while quality control mainly focuses on test and verification activities at late stages. In the product development process, Q-gates is a type of stage gate, embedded in the product development process, where product reviews are held to ensure that quality targets are expected to be met.

FMEA

Failure Mode and Effects Analysis (FMEA) is a method for predicting the potential failure modes of a product during its life cycle, the effect of these failures and the criticality of these failures for product functionality [2]. The main goal is to limit or avoid risk. The method has its origins at NASA, where it was initially developed in order to detect the possible potential failures at the design stage [3]. It was later implemented in a number of industries, mainly in the car industry, aerospace, electronics and nuclear industries [4]. The method is more closely related to reliability engineering than quality engineering. In quality engineering, the product either passes a given test or it fails. On the other hand, reliability is usually concerned with failures in the time domain [5]. Both of the Swedish companies visited and one of the Japanese companies use FMEA.

At one of the Swedish companies, the analysis for the main module was used as a reference when performing the FMEA analysis on variants. This was perceived as very advantageous in order to save time. It was also stressed that the most experienced and knowledgeable people should do the main module analysis, making it a design support for other groups analyzing variants. Alongside the FMEA, describing failure based on components, the connections in between functions were mapped in a diagram. It is often recommended to conceptually analyze robustness in the early stages of product development [6]. Further, the sensitivity of these functions or properties to different types of disturbances was identified. Properties could be ergonomics, assemblability or performance. Disturbances could be piece-to-piece variation, change over time, user, environment or system interaction.

At the other Swedish company, a “best practice” database for knowledge capture was under development. This was made to ensure that knowledge was not lost when changing employees and that knowledge was transferred between projects. The system was to consist of guidelines for product development and production-related aspects, along with a number of checklists. One aim of implementing best practice system was to be able to identify problems in early development stages. The companies had previously begun working with both process and design FMEA. An aim was to integrate the FMEA methods with the best practice system. This system should also be a support for requirements specification and requirements follow-up activities.

CAx

With the advances of computer technology, many product development life-cycle processes have been automated by the introduction of computer-based systems, such as Computer-Aided Design (CAD), Computer-Aided Process Planning (CAPP), Computer-Aided Manufacturing (CAM) and Computer Aided Tolerancing (CAT). Many computer-based concurrent engineering design systems have been developed to further improve design quality. They do so by considering down-stream product development life-cycle aspects (such as manufacturing, assembly, maintenance, recycle/disposal, etc.) at the early design stage.

Traditionally, Japanese companies have used in-house, computer-aided tools in the design process. The reason was that a greater control of the design could be achieved. Many Japanese companies had experienced that the quality of their products had decreased since they started to use commercial CAT tools that had been the major motivators for using in-house CAT tools.

One Japanese company mentioned that since they rely heavily on the usage of CAx tools in the early phases of product development, it is important for quality that these tools give accurate results. However, the company does not evaluate the effects of geometric and material deviations in CAx software. The biggest error in simulation, they say, does not stem from insufficient material models, variation in geometry or software error; rather, it stems from the boundary condition errors. Software is constantly benchmarked against results from testing. These results are then analyzed probabilistically to calculate a Process Capability Index (C_{pk}).

Conversely, at one Swedish company, geometric and material deviations were addressed in the CAx software. This was done by using tolerance analysis functionalities in the existing CAD system. These activities were the responsibilities of the individual engineers. Wider-ranging robustness methodologies were not prioritized. Six-Sigma was discussed, but the product volumes were considered too low for that methodology.

Quality Supporting Activities

During the interviews, several methods and tools supportive to the work with quality were identified. Some of these are part of what we define as Lean Manufacturing and Lean Product Development.

Kaizen

Like most Japanese companies, the ones we visited in Japan all subscribe to the philosophy of Kaizen (Japanese for “improvement” or “change for the better”). In it, small incremental changes to a product are preferred instead of analyzing and redesigning whole systems in one step. Engineers should focus on improving what is faulty, instead of “fixing things that are not broken.” Good product quality comes naturally with well-tested and reliable product designs. The Kaizen philosophy was also identified at one of the Swedish companies visited.

Front-loading and Incremental Changes

One of the Swedish companies considered having the most experienced people performing the robust analyses an important precondition. Extensive project front-loading was also imperative. Another reflection this company had was that in order to implement robust design, a Japanese approach to product development was required. This approach involves incremental product changes instead of radical innovation. When performing robust design analyses, if a new product (or, as in this case, a variant) is not based on an older, already manufactured product, sufficient knowledge or data does not exist to provide necessary input to the method.

In the same way, one of the Japanese professors discussed the process of securing high product quality. He pointed out the importance of putting the right resources in the right place in the development chain. He especially mentioned one point: during the early phases of the product development process, the project selects the very best of engineers (for this phase). This, he means, secures a good starting point for further development. And as many quality-related issues can be identified and eliminated at low cost here, relative to changes executed late in the project, this saves both time and money.

Standards, Guidelines and Databases

Something used in Japan, as well as globally, is standardization, the process of developing and agreeing upon technical standards. A standard is a document that establishes uniform engineering of technical specification, criteria, methods, processes or practices. By using standardization, it is easier to communicate through the set of guidelines. Standardization also links to Lean Manufacturing, which is a known method developed in Japan. The Japanese companies seemed to strictly follow the standards and guidelines implemented in the daily work. At one of the Japanese companies, these guidelines (with their related checklists) were inherited from previous products and then updated continuously. We were shown large, A1-format spreadsheets containing guidelines and checklists. These sheets were referred to as Design Stories. Although the text was mostly in Japanese, we were impressed by the level of detail that was fitted on a single piece of paper, albeit large.

Limit trade-off

One of the Swedish companies used limit trade-off curves as a quality supportive tool. When using them for robust design purposes, combinations of design parameter levels are varied until product failure occurs. The main idea is to find the levels for failure instead of doing experiments where parameter levels are just varied near the nominal values. The aim is to find a design where the "window" in which the product still operates is as large as possible.

In one Japanese company, we also identified a philosophy of focusing more on facts and numbers. Quantitative answers should always be preferred to qualitative. Instead of answering “The stress levels are o.k.,” the answer should be “The stress levels are 145.50 MPa.”

Modularization

At one of the Swedish companies, it was decided that if a Robust Design Methodology had to be implemented, it might as well be useful. They combined the methodology with their modularization strategy. Since their product was customized to suit each of their customers, a main module had been designed, from which all other variants were derived. The robustness analyses were initially performed on the main module, thereby providing a basis for the analyses performed on the variants. Analyzing robustness had thus become less time-consuming. It was also described that the further away a variant was from the main module, the greater the risk in that project.

A key prerequisite at this company was that design methods applied should not be too time-consuming. By connecting the design methods to modules, the time spent was minimized and method results could be reused. Among the advantages of implementing the methods was gaining an overview of noise factors on a parameter level and the documentation and knowledge transfer that was achieved as a consequence.

Customer focus

The experts interviewed in Japan all had long industrial experience, and one of them emphasized that many Japanese companies assure that the right product is being developed by extensive focus on the customer. Having well-defined customer needs will support the process of "doing the right thing," which is an important factor for guaranteeing good product quality in the end [6]. This might not seem very remarkable. However, interesting in what he said was that customer needs were often identified using the employee or employee relatives as "test persons." The fact that employees were chosen for the identification of customer needs might seem to be questionable since they might be biased. Mass media was identified as another input source for customer needs.

TRIZ and QFD

Other quality supportive methods encountered in this study were TRIZ and QFD. TRIZ is a Russian method for problem solving and analysis and is a forecasting tool derived from the study of patents of inventions in the global patent literature. It was developed in 1946, and is rendered in English as “the Theory of Innovative Problem Solving.” According to one of the professors, around 10% of the Japanese engineers are familiar with TRIZ. One of the ideas is also to transfer knowledge from existing products to the new one that is to be developed. It is also a method to be used early in the design process. QFD is a method to transform user demands into design quality and was developed in Japan during the 1960s. QFD is designed to help engineers focus on characteristics of a need or existing product or service from the viewpoint of market segments, company or technology development needs.

Kepner Tregoe

A method briefly discussed by one of the Japanese companies was the Kepner Tregoe Rational Process Technique. Developed by the American management consulting and training service company with the same name, it is a method that could be likened with a combination of Fault Tree Analysis, Ishikawa diagrams, Root Cause Analysis and The 5 Whys.

Summary

One main impression from this study is that while Swedish companies are inspired by Japanese product development methods written by American authors, Japanese companies are partly influenced by American methods. They are not familiar with all the “Japanese” design methods used outside Japan. There is today an interchange of ideas between countries, and it is no longer easy to identify approaches in different countries as isolated phenomena.

One difference between Sweden and Japan is that Japanese companies have special prerequisites for robust design and quality assurance based on their engineering practice. There is a tradition of continuous improvement in product development. There is also a tradition of keeping track of product and production data that can be used for quality assurance in product development. This was confirmed during company visits and expert interviews. The typical “Swedish” way of developing products is more difficult to describe. Sweden has traditionally been inspired by European and American schools of thought, which have had great influence on the terminology and methods used. Cultural prerequisites for product development have gained less attention.

A main trend in both countries is a striving to formulate a process for robust design or quality assurance activities. These can be compared to programs promoted under the name of “design for six sigma.” A common theme is to provide a holistic process for quality assurance throughout the product lifecycle. There is great variety in the formulation of these different processes. Introducing tools for this through programs such as design for six sigma can be a means of achieving change in an organization. It seems, however, that companies often have the ambition of formulating their own quality assurance process, putting together a selection of commonly known methods such as QFD, FMEA or the P-diagram. This has the advantage that the implementation becomes more pragmatic and more adapted to specific company prerequisites. It is also a benefit that engineers can participate in formulating their own work procedures.

Visiting Japan, it is difficult not to be impressed by the Japanese meticulousness and work discipline. Apart from employing appropriate product development methods, hard work and a perfectionist approach to work tasks must certainly be factors explaining the success of Japanese companies. In Sweden, the Kaizen way of thinking is a popular way of transferring this mindset. The Japanese tradition of moving from the detail to the whole has some advantages compared to the western way of drawing up abstract analyses to analytically formulate an approach to a problem and then fitting in the details. Even so, Swedish companies could benefit from carefully evaluating new methods to ensure that they are truly appropriate for the Swedish culture.

[1] Phadke, M. S., "Quality Engineering Using Robust Design", Englewood Cliffs, New Jersey: P T R Prentice-Hall Inc., 1989.

[2] Teng, S.-H. and Ho, S.-Y., " Failure mode and effects analysis: An integrated approach for product design and process control", *International Journal of Quality and Reliability Management*, Vol. 13, 1996, pp. pp. 8-26.

[3] Puente, J. and Pino, R., "A decision support system for applying failure mode and effects analysis", *International Journal of Quality & Reliability Management*, Vol. 19, 2002, pp. 137-150.

[4] Sharma, R. K., Kumar, D. and Kumar, P., "Systematic failure mode effect analysis (FMEA) using fuzzy linguistic modelling", *International Journal of Quality & Reliability Management*, Vol. 22, 2005, pp. 986-1004.

[5] P, T. D. O. c., "Practical Reliability Engineering", John Wiley and Sons, 2002.

[6] Andersson, P., "A Semi-Analytic Approach to Robust Design in the Conceptual Design Phase", *Research in Engineering Design*, Vol. 8, 1996, pp. 229-239.