AT THE END OF THE WATER-FALL

Author: Susanne Drexl-Wittbecker, ROI Management Consulting AG

OVER THE NEXT FIVE YEARS, 90% OF ALL INDUSTRIAL COMPANIES WILL OFFER PRODUCTS THAT HAVE A SOFTWARE COMPONENT AND ARE NETWORKED IN THE IOT.

They form the basis for product-related services and pave the way for new, digital business models. Only the development departments are not prepared for this.

When Hurricane Irma hit the Florida coast in September 2017 and hundreds of thousands of people left their homes to flee inland, electric car manufacturer Tesla responded immediately with a software update. With it, the manufacturer deactivated the power limitation of the battery used in some models and thus released additional battery capacities. A simple download increased the range of the vehicles by around 20% in one fell swoop and enabled thousands of Tesla drivers to safely escape hurricanes.

REACH AS A SERVICE

Tesla's example symbolizes a new product world in which classic industrial goods are increasingly defined by elements beyond hardware. The combination of electromechanical components, computing power and connectivity forms the basis for a range of complementary solutions and services for the core product, such as predictive maintenance or the leasing of machine capacities. These smart products enable manufacturers to stay in contact with their customers beyond the point of sale, add functions and thus adapt their products to current trends - and monetize them. This not only changes the functional scope of individual products, but also the entire product portfolio of a company. Instead of a product that is sold once and used in the same way by the customer for 20, 30 or even 40 years, a hybrid bundle of physical components and complementary applications that can be offered to the customer throughout the life of the product occurs. Tesla's example makes it clear how this works in practice: because

the limitation of battery performance is not primarily for technical reasons, but is part of the price model: customers can purchase a cheaper vehicle variant that does not differ from the other models in terms of hardware, but can only access 60 kWh of the built-in 75 kWh battery power. So it is no longer the hardware that forms the distinguishing feature between the variants of a model series, but the software setting. Range, engine power or pollutant emissions are no longer characteristics acquired with the product, but additional services. Welcome to the age of Smart Services!

SERVITIZATION OF THE INDUSTRY

This development does not only affect the automotive sector, but covers almost all industries, from consumer-oriented electronics manufacturers to automotive OEMs to mechanical and plant engineering. According to a recent ROI survey, 83% of companies believe that their products can be digitally enhanced (cf. DIALOG 50 and 57). The increasing shift in revenue potential in the direction of digital additional services is putting manufacturers of classic industrial goods under considerable pressure. On the one hand, because so far their organizations, processes and methods have generally not been designed for the dynamics and changed cycles of digital products and services. On the other hand, because more and more new competitors are appearing that do not originate in the manufacturing of hardware, but try to roll up the market of traditional manufacturers via software or business model features.

WELCOME TO THE AGE OF SERVITIZATION

Fig. 1

The development department is at the centre of these market dynamics: on the one hand as a driving force in the identification of product-related services for changing customer requirements. On the other hand as enabler for new services and business models in terms of a fast and efficient implementation "from idea to market". In practice, however, most development departments are not at all geared to these requirements in terms of personnel, processes and organization. To prevent the development organization from becoming a bottleneck on the way to becoming a smart products provider, companies must adapt their processes to the logic of digital products and services. This requires a new, smart development approach, or in short: Smart R&D.

THE SMART R&D ORGANIZATION

This development approach is characterized above all by the fact that it is radically oriented towards the customer and his product requirements, which can be incorporated into the development in all phases of the product development process. The entire organization, the processes, the employees and their mindset must be designed in such a way that customer requirements can be incorporated or changed at any time. This requires high flexibility and agility in the processes and structures on the one hand and the ability and appropriate methodology on the other to quickly and efficiently convert such requirements into functional and testable solutions. To achieve this, essential aspects of the development organization must be rethought:

REQUIREMENTS-ENGINEERING

FROM PRODUCT TO ECOSYSTEM PERSPECTIVE

Henry Ford's much quoted aphorism that people, when asked what they wanted most, answered with "faster horses", summarizes well the challenges of modern requirements management. Especially in the context of smart products, the question of what a product should be able to do has to be viewed differently than it has long been the case with classic, industrially manufactured products. Digitization will considerably broaden the range of potential product functionalities. However, many of them do not result directly from the core product itself, but from the way it interacts with other products or infrastructures in a complex ecosystem. For example, when vehicles communicate with each other or with the traffic infrastructure in order to reduce the risk of accidents.

The essential task of requirements engineering is to identify these (mostly implicit) customer requirements, to evaluate them and to convert them into concrete functional scopes. On the one hand, it is important to consider the entire product life cycle and to consider where additional innovations can be introduced meaningfully in the form of a software solution or a service. And on the other hand, to take into account the external conditions, such as platforms or operating systems, which they access. The development organization must therefore not only keep an eye on its own customers, but also on other providers and their systems. To break down this complexity into concrete requirements, to synchronize the different life cycles of the individual partial solutions and to define interfaces are activities that require considerably more attention than in the past and are hardly sufficiently mapped in the classical development models so far (see Fig. 1).

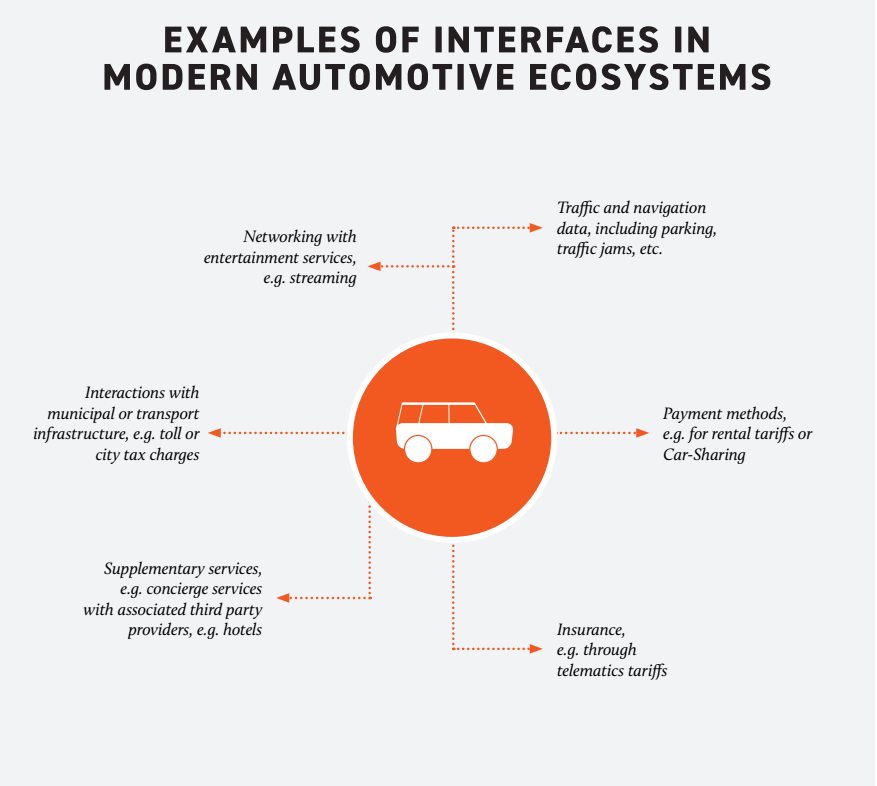

Fig. 2

In order to do justice to this, smart development models therefore rely on an early and continuous dialogue with the customer. Methods such as Design Thinking are used to attempt to take into account the perspective of the design from the outset in the development process. Early and regular testing via MVPs and rapid prototyping also ensure continuous customer feedback throughout the entire development process. Allignment with corporate management also represents a key success factor in the development of smart product solutions. Beacuse the question of which revenues are to be generated in the future with the core product and which with product-related services directly affects the long-term strategic orientation of the company and should therefore be integrated into an overarching portfolio strategy that is also supported and consistently supported by the company management.

PROCESSES

FROM WATERFALL TO HYBRID DEVELOPMENT MODEL

Additional product functionalities, especially in the software area, not only increase the development effort, but also create completely new roles and tasks in the development process that must be integrated into the overall process. In particular, the synchronization of the various functions in hardware and software development, which each have very different development cycles and working methods [see info box], poses a particular challenge in the development of smart products for which most development departments do not have suitable control models. Hybrid development models such as ROI's (see article on page 12) provide an approach for integrating agile methods with the classic Stage-Gate approach, creating regular points of interaction with customers (see Fig. 2).

In addition to controlling at the project level, it is also important to build up digital solutions and supplementary service offerings across various product groups in a modular manner within the framework of systematic portfolio management so that the internal value chains can be streamlined.

ORGANIZATION

FROM PROJECT TO PRODUCT THINKING

Hardly any other organizational unit in a company is exposed to such massive changes as the development department. The expansion of the product portfolio to include smart products and services not only leads to a high demand for new specialists, but

also changes the type of cooperation within the development organization.

Whereas the development process used to end with the start of production, smart products are now being continuously further developed with the help of updates and functional enhancements. This shift in the product development process well into the market phase means for the development organization that its capacities are tied to a product for longer than before. The classic project organization, with a defined start and end point, can no longer be applied to this. Instead, development departments will have to think more in terms of products or product groups and releases, similar to software manufacturers.

This also includes networking the development department with other indirect areas such as product management, maintenance or customer service in order to accompany a product throughout its entire service life and to incorporate the findings from the other areas in its further development.

IS SOFTWARE EATING THE R&D?

The increasing proportion of software in numerous industrial products is already changing the processes, organizations and working methods in the development organization. Many of these adaptations have their origin in the agile working methods of software development, which are now increasingly finding their way into hardware development. With their high transparency of results and flexibility, they offer the ideal set-up for smart product development in many respects. Nevertheless, an approach aimed exclusively at adopting agile methods does not go far enough.

On the one hand, because a smart development process cannot be reduced to individual methods such as SCRUM or Kanban, which can never completely map its complexity. Rather, it is about the entire organization, the processes, the employees and their mindset being aligned in such a way that customer requirements can be incorporated or changed at any time. On the other hand, because certain elements from the classic hardware development process, such as checks and validations, will continue to be necessary. These security levels in particular are aspects in which software development can learn from hardware development. Compared to pure software products, compliance with norms and standards plays a much greater role in combined smart products - for example when it comes to combining hardware and software in the field of autonomous driving (see Fig. 3).

And finally, companies must be careful not to overburden their employees. Often hardware developers are not used to the high frequency and transparency of results. These "cultural differences" often lead to a frightened defensive or blockade posture. Managers are therefore challenged to shape this transformation process in the development department ... and to accompany it consistently by coaching and reducing fears on the one hand, but also by allowing more autonomy in the teams on the other (see page 24).

DIFFERENCES IN HARDWARE AND SOFTWARE DEVELOPMENT

HARDWARE		SOFTWARE
STAGE-GATE APPROACH	VS	AGILE PROCEDURE
LONG-CYCLE DEVELOPMENT TIMES (WEEKS/MONTHS)	VS	SHORT CYCLE DEVELOPMENT TIMES (DAYS)
LOW COMPLEXITY	VS	HIGH COMPLEXITY
GOAL: ERROR-FREE DELIVERY	VS	GOAL: FAST DELIVERY
RIGID STRUCTURE	VS	FLEXIBLE STRUCTURE

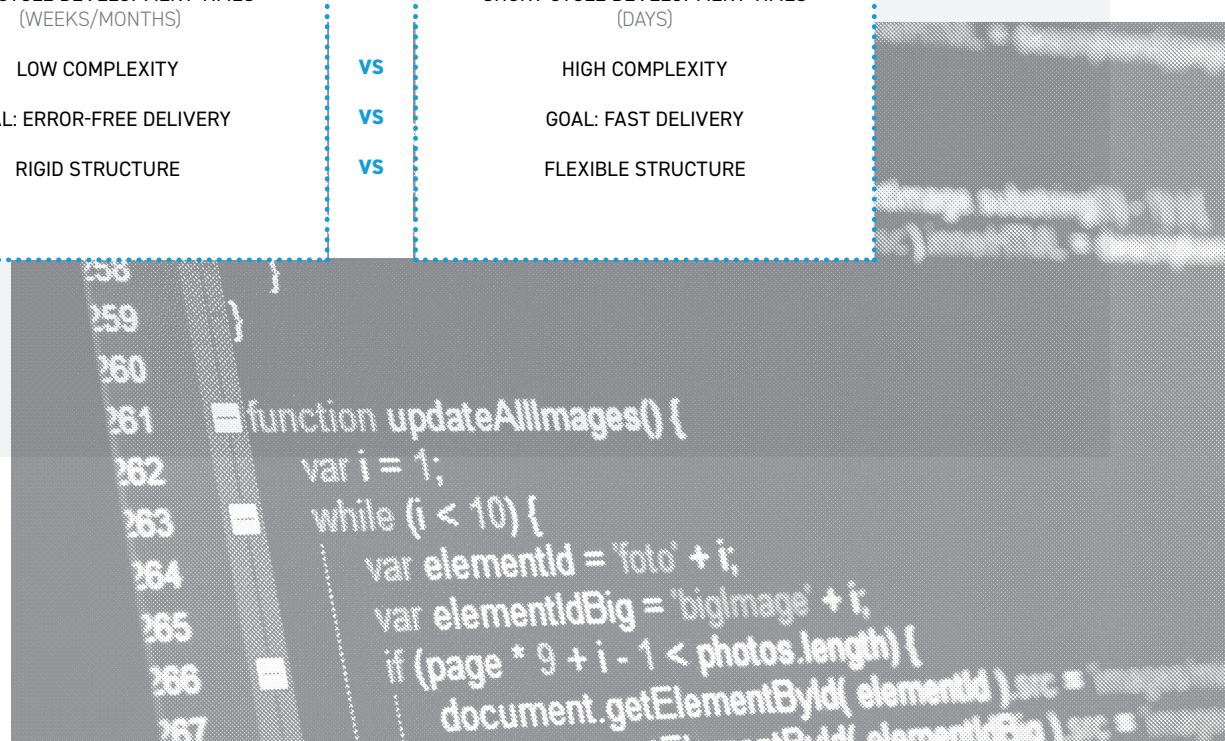

Fig. 3

BOTTOM LINE

Companies that manage to overcome these tensions between the old and new R&D worlds and to combine the competencies from the hardware and software sectors in the best possible way certainly have considerable competitive advantages here. In the age of smart products, an agile development organization that is able to react quickly and flexibly to dynamic customer requirements becomes a strategic success factor for entering digital business models. The example of Tesla shows once again how difficult this process can be sometimes. This is because Californians regularly fail to transfer their "digital-centered mindsets" to the supposedly outdated structures of physical vehicle construction, as is evidenced by delayed deliveries and constant quality problems.