Whitepaper

SCRUM IN HARDWARE DEVELOPMENT

Engineering expertise is the export hit of German, Swiss and Austrian enterprises. Development-intensive branches such as mechanical engineering, the automotive industry, medical technology and electrical engineering are key mainstays of the economy across Europe. However, global competition is catching up: shorter innovation cycles are called for, development projects must be completed faster, more economically, but nevertheless, in the same high quality. Agile management frameworks, such as Scrum, uncover as yet unexploited potential

Switzerland and Germany take top places in international innovation rankings as clear leaders of technology: Switzerland, for example, ranks No. 1 in both the Global Innovation Index (GII) and the Innovationsindikator des Bundesverbands der Deutschen Industrie (BDI- Federation of German Industries).[1] In the GII Germany took 13th place and 6th place in the BDI-Innovationsindikator. But changes are on the horizon.

[1] Deutsche Telekom Stiftung (publisher): Innovationsindikator – A comparison of the innovation ability of the most important industrial countries. www.innovationsindikator.de

The Global Innovation Index 2014: The Human Factor in Innovation. Johnson Cornell University, INSEAD, World Intellectual Property Organization. Editors: Soumitra Dutta, Bruno Lanvin, Sacha Wunsch-Vincent. http://bit.ly/1uibjl0

"We know that the nation that goes all-in on innovation today will own the global economy tomorrow", said President Barack Obama, not without reason, in his State-of-the-Union speech on 28th January 2014. According to the findings of a survey undertaken by the Verband Deutscher Maschinen- und Anlagenbau, China is all set to take on the role of innovator – among other places in the European (and most especially German) export domain of mechanical engineering.[2] To begin with its own market is to be supplied more independently of foreign and often too expensive products. Research and development in the other BRICS states is also aimed at finding technologically attractive, but nevertheless more favourably priced solutions for their own economic areas and other developing and emerging countries – with export to the industrial nations not excluded. The BDI ranks China at place 22, in other words three places higher than in the year 2000. China invests in education and training: around 7 million young people graduate from universities a year, seven times more than in 1999, almost a third of whom are engineers. In addition Chinese enterprises plan investment targets abroad, in order to gain know-how.

[2] VDMA: China is a strong competitor and driving force for German mechanical engineering. Press release dated 18.2.2014. http://bit.ly/1kQo1W7

Such fundamental changes on the market are forcing European hardware manufacturers to develop faster, more efficiently and in a more integrative manner than ever before.

- **Faster development**
  Innovation in development is and remains the mainspring of success. The faster development can be achieved, the faster a company is able to respond to changes on the market.

- **More effective development**
  Where fast innovation is concerned, "cost cutting" is not a sustainable approach. An adequate strategy must be adopted to ensure that the development itself generates cost benefits. This sometimes entails a break with sequential development processes.

- **Integrative development**
  The growing complexity of hardware, especially with the integration of software in embedded systems, requires an interdisciplinary and cross-functional approach in development. Colleagues from different domains work together in a team, mutually coordinate their development cycles in an expedient manner and collectively develop the integrated product. In addition international competiveness today demands collaboration in teams all over the world.

When engineering rescued software development

So the development of physical products is today confronted with the same problems as those of software development 20 years ago:

- High cost pressure
- Time pressure
- Extreme specialisation on the one hand, but increasing demands on the complexity and possibilities of the products on the other
- The use of simulations in virtual space, in order to make work with prototypes more cost-effective.

Let's take a look back into the past: "agile" management frameworks, such as Scrum, emerged at the end of the 1990s, because faster and cheaper deliveries were demanded. The approach was not new: "Iterative and Incremental Development" (IID) was being used as early as the 1940s to ensure fast development of fighter jets (e.g. the XP-80 and X-15).3 Decisive for the rapid establishment of agile development in software was the awareness that: the development of software is always the development of something new – and what is new cannot be specified in detail in advance.

[3] Larman, Craig; Basili, Victor R.: 3 Iterative and Incrementai Development: A brief history. In: IEEE Computer, Volume 36, Issue 6, June 2003, S. 47-56.

Works by Hirotaka Takeuchi and Ikujiro Nonakahe constituted the rationale of agile management frameworks. In their article "The New New Product Development Game"4 published in 1986, they analysed the innovation processes of Fuji-Xerox, NEC, Canon and Honda. Their findings led them to the conclusion that high productivity and innovation in these companies were the result of the transfer of knowledge in self-organised, cross-functional teams. At the same time the awareness for lean management, team work and just-in-time processes emerged in the manufacturing sector, based on the Toyota production system. Both developments have the same conviction in common: **a challenging goal** and **iterative development cycles** combined with the emerging team dynamics that arise from the **commitment** to continuous improvement, set in motion a cycle of knowledge creation and the delivery of products, at the end of which there is found not only **innovation**, but also **a faster time-to-market** and **higher quality..**

[4] Nonaka, Ikujiro; Takeuchi, Hirotaka: The New New Product Development Game. In: Harvard Business Review, Jan-Feb 1986.

Agile methods have become the standard in software development and, as the study "Status Quo Agile 2014" from the University of Koblenz indicates, they are also spreading outside of IT. 27% of those interviewed stated that agile methods are meanwhile being used in sectors not connected with IT. 86% found **Scrum** to be the most frequently used agile method.[5]

[5] BPM-Labor der Hochschule Koblenz, Prof. Dr. Ayelt Komus: Status Quo Agile 2014. Second study on the use of agile methods. In cooperation with the GPM - Deutsche Gesellschaft für Projektmanagement and the International Project Management Association (IPMA)

Scrum

Scrum is a framework for iterative and incremental – "agile" – development in which a product is refined step-by-step in cycles from the vision onwards. People know exactly what they want, but accept the fact that they cannot specify the product right down to the last detail. In order to reduce complexity, a Scrum team works in individual steps of maximum two to four weeks – so-called "sprints". To start off with the key functionalities are determined and then developed further or rejected in each sprint on the basis of customer or user feedback. Unlike classic project management, the customer does not have to wait until the end of the entire development to see the result, but is instead involved in every step of the development. The organisational principles of Scrum:

- **Small, self-organised, self-responsible, cross-functional teams** – consisting of the ScrumMaster, product owner and development team.
- **Working in compliance with the pull principle:** the development team decides on the number of functions (that have been prioritized by the product owner based on the business value) it will develop in a sprint and also decides on how it will carry out the related tasks.
- **Clearly limited time intervals (timebox):** the aim is to actually finish the selected functions within the sprint.
- **Useable business functionality:** at the end of each sprint the team has to make a delivery, which complies with the standards, guidelines and prerequisites of the project.

A key factor in the success of Scrum is intensive communication: the team members confer on the current state of the work in short daily meetings (daily Scrum). They work on the joint conception of the functionalities to be developed in estimation meetings (backlog grooming) and in the sprint planning meetings 1+2 specify what, when and how development is to be carried out. In the review meetings at the end of a sprint the team talks with the customer about the developed product increments and retrospectively assesses its work in order to improve this where necessary.

How hardware development benefits from software development

Some special features should be taken into account when selecting agile management frameworks for hardware development:

- **Changes:**
  The components of a hardware system must function both individually and jointly. Changes are often expensive, especially at the end of a project. The later systems are integrated, the more difficult changes become.
- **Dependencies**
  The dependency of suppliers is extremely high due to the need of materials, tools and components. The flexibility and speed of the suppliers has a great impact on the development process. That is why the depth of integration is of great significance and interaction based on partnership so important.

- **Regulations**
  Strict quality and safety regulations must be taken in account in engineering. This has an impact on documentation, among other things. In this point software development shows that the quality of documentation in agile projects wins vis-à-vis waterfall projects. Regulations concerning documentation are adhered to, but with an enhanced understanding of what is expedient and really important.

- **Prototypes**
  The construction of prototypes for verification and validation of the development entails many work steps and great expenditure of resources. So creating the right development and integration environment is therefore decisive. What, for example, can be tested virtually? Although at first sight this increases the development costs, it is more than compensated for by benefits in speed

- **Constraints**
  The definition of framework conditions (constraints) is particularly critical and important in the development of hardware. The size of a PCB, for example, determines the available processor capacity. In the case of embedded systems, this in turn also has repercussions for the software development. Constraints must be taken into account in the design of a product from the very outset. In regulated environments especially, these limits are a question of safety and are not negotiable.

Despite these differences to software development, more and more enterprises from the "traditional" hardware sectors are coming to recognise the value of agile development for their own competitiveness. In this they profit - in adapted form - from practices that have been mapped and refined in agile software development over the last 20 years:

- **Continuous delivery**
  The consistent iterative delivery of product increments (these can be CAD drawings, for example, in hardware development) immediately generates feedback. Development teams can use this to design ever more detailed solutions in line with the production vision and the constraints and thus use their resources in a focussed manner. Continuous feedback means that miscalculations can be rectified immediately - this both speeds up the development projects and makes them easier to control

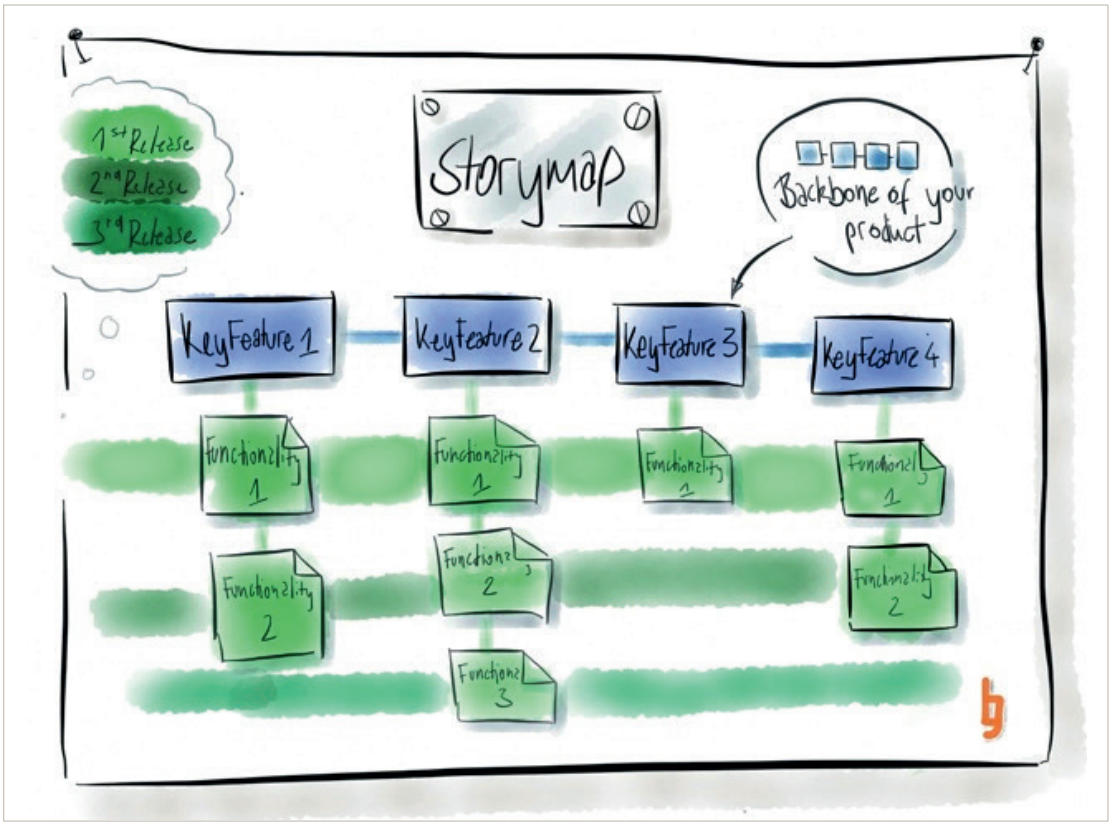
- **Making planning, progress and problems visible in the process**
  Simple procedures have been developed for agile management frameworks, which help make project planning visible for cross-functional teams:

  - **Backlog** – a list of clearly distinct requirements for the product

  - **Story maps** provide the systematic overview – to enable the coordination and integration of the component development by several teams (see diagram on the next page).

  - **Meeting formats** – ffor daily coordination and regular, targeted feedback on the progress of the project in short time intervals.

  - **Reporting tools,** that quickly show teams where they stand.



Schematic diagram of a story map

- **" Test first" approach**
  It is possible to test the results during the development process even in hardware development – especially if CAD technology is used. Nevertheless, it is still necessary to be able to work as quickly and economically as possible on real prototypes in hardware projects. It is expedient here to involve the company's own workshop, instead of just leaving it to repair what the supplier has developed.

- **Work in cross-functional teams**
  You can see it particularly well in hardware development: once the prototype is finished and all the engineers are swarming around it – that is when the magic happens. Touching it, turning it around, inspecting it from all sides and experiencing it with all senses ignites the creative spark – the energy for new ideas is there! But in reality it is more often the case that less and less work is carried out in teams, instead it is increasingly left up to specialists to do. Development processes based on a strict division of responsibilities gives rise to many, expensive handovers. That is why Scrum relies on clearly defined, cross-functional teams that overcome such division of responsibilities and are therefore able to develop faster. Such teams are made up of all the persons necessary for the development of the product – from the buyers and suppliers to the electronic technicians, software and firmware developers and on to include assemblers. They should all see and understand how the components interact with each other.

Real life examples: Hardware development in laboratory diagnostics

1. Cross-functional collaboration in the construction of prototypes

In order to promote collaboration the team works in a factory workshop that has been especially reserved for the project. Development is carried out in one part of the workshop, while in another part devices are constructed and tested. The different departments can no longer be distinguished: the ScrumMaster and product owner both have their workplaces here, the CAD designer sits next to the technical drawer and the mechanic, while the assembler from production works a few metres away on the construction of the prototype. All information concerning the project converges in the "Scrum Arena" – an area of two by three metres in which all the Scrum artefacts relevant for the everyday work on the project are found: Task Board, Product Backlog, Release Plan and Impediment Backlog.

On a typical working day the first 15-minute Daily Scrum on the construction of the prototype takes place at 9 a.m. Not only people from Assembly and Development are present, but also from Purchasing, Logistics and Technical Documentation: together they discuss how the team is progressing with the goals of the week: Where are parts missing? What deliveries are scheduled for this week? What errors have been noticed during assembly? Where is external support needed? The hardware team meets next, followed by the software team - always in the same arena. There are never only people from software present or only from hardware – at least one representative from the other discipline always takes part. There is a review meeting every two weeks, during which the team uses the prototypes to demonstrate the current status of the development.

2. Production increments with user stories

A module for reclosing test tubes is to be implemented for a laboratory diagnostics device. Initially the product backlog only contains the following requirement in the form of a user story: as a medical-technical assistant I do not wish to close the test tubes by hand, because I do not wish to overstrain my wrists..

Unlike a classic requirement, the user story does not say anything about the properties of the system; instead it is limited to the problem to be solved. This prevents an overhasty commitment to solutions – the focus of attention is on understanding the problem and the desired use. The user story is analysed immediately before implementation in the sprint planning. The user story is specified with the development team:

- **Requirements**
  The test tube is to be closed by automatic pressing on of a cap.
- **Acceptance criterion (validation)**
  The MTA no longer has to close any test tubes by hand.
- **Acceptance test**
  600 open test tubes of all permitted types are closed within 30 minutes.
- **Constraints**
  - Must function with test tubes 13-16 mm in diameter and with heights of 65-100 mm.
  - Cap must be leakproof.
  - Cap must be made of polyethylene.
  - Cap must be blue.

new functionality complies with the requirements. Furthermore: customers, users and other participants close to the market (e.g. field workers, Marketing) assess in the review meeting whether or not the implemented solution fulfils the intended use.

3. Taking constraints into account during system development

Constraints are non-functional requirements on a system or component level that define the characteristics of the product. One example here is the definition that a transport system must transport and process laboratory specimens in the form of test tubes of a specific size. This constraint is significant for the characteristics of the product, so that it must be taken into account in virtually every production increment, either directly or indirectly.

Another example is the goal of increasing the throughput speed during the transport of specimens from 2000 to 3000 an hour. On the system level this constraint demands the transport speed of the specimens for all modules (input and output, transport, analysis, etc.) with the overall throughput as the goal. On the component level the constraint demands optimisation of the transport speed for each of the individual modules respectively: input and output, transport module, sampling etc.

The constraints on the component level prescribe the development path: the speed of the individual modules can be optimised by degrees and then tested in the overall system.

The constraint is always the same: 3000 specimens an hour. But the level at which this goal is to be achieved varies. There is a difference between whether the modules achieve the desired speed individually, or whether the overall system does this.

3 good reasons for Scrum in hardware development

1. **Delivery of what the customer really needs**
   The scope of functions and the suitability for use are decisive for success on the market. Although user requirement specifications define the wishes of customers and users, they do not necessarily always reflect what is really needed. Incremental delivery enables continuous validation by the customer and helps you to develop what your customers really need. Expensive miscalculations are detected early on and can be rectified.

2. **Faster on the market**
   If you deliver iteratively and incrementally, you develop the key functions at the beginning of the development process. This could lead to a product having so much added value for the customer at an early stage, that some of the originally planned features become obsolete.

3. **Keeping within budgets**
   If you deliver iteratively, you deliver quickly and reliably. So there is no danger of ending up with only a semi-finished solution, despite the budget being used up. Agile development enables a clear and simple control of costs. The risk of investing a lot of money in solutions that are unusable at the end of the day is minimised.

Start off with: what you should pay attention to when selecting a consultant
The development of hardware using Scrum is still relatively new – this makes the market all the more attractive for consultants, because a trend is actually developing. There are currently only a few specialist providers in the German-speaking area that have more than five years of in-depth experience with Scrum, not only on the team level, but also in scaled environments. There is, of course, even less experience with the development of hardware.

VMany providers concentrate simply on implementing the "Scrum handcraft". A Scrum implementation with long-lasting success is, in fact, the task of forward-looking change management. Scrum does not only have an impact on the collaboration in teams, but also on the interfaces. Questions of management and self-management arise that require knowledge and experience in the handling of social dynamics. So Scrum is not simply an add-on to classic project or requirements management, but an independent discipline in its own right. **Agile development thinks differently and acts differently.**

Your consultant should not only be well versed in Scrum and the necessary development practices and bring in experience from other projects, but should also be innovative and agile in his/her own approach to a problem. Below our checklist for selecting your consultancy team:

1. Has the provider already consulted customers in different branches of industry?
2. How much experience do the individual Scrum consultants have?
3. What role does the provider play in the agile community?
4. Do the individual consultants have experience in cross-company collaboration in an agile context, for example with suppliers?
5. Can your consultant show successful scaling projects with more than 60 – possibly even globally distributed – employees?
6. Is uniformity guaranteed, because all consultants of the provider work in compliance with the same principles?
7. Are you told what you want to hear, or are the diagnoses painful sometimes
8. Do you see the provider and his/her colleagues at conferences?
9. It is possible to see how Scrum works in practice at other customers of the provider
10. Are the consultants the type of persons who feel responsible for the success of your company?

How does borisgloger consulting work?
Your organisation is already unique. Agile product development preserves this uniqueness while at the same time preparing it for changes on your markets. So it is our aim to render ourselves unnecessary: you yourself and your employees are the ones that must live agility – we cannot do that for you. But we can build the necessary foundation: from the very outset we use simple images, methods and clear processes to create a common understanding among all those involved – from the development team to the management - of what agile methods mean for the organisation. Based on this common understanding we establish pilot teams consisting of members from all the divisions and departments involved, who decide themselves how they should start with the tasks necessary for the implementation of Scrum and how they wish to go about solving these. This includes setting up a first Scrum team: this team runs through several sprints during which it collects information about what can be improved together.

The management is involved from the very outset and takes an active part. It makes the necessary decisions and itself learns what it means to work with Scrum. This integration is very important for us, because trust in the new methods must be built up on all levels. During a Scrum implementation you will notice that the activity profiles of employees changes over time – but not in compliance with prescribed definitions. The employees create their new roles themselves. We integrate appropriate development practices step-by-step during the work with Scrum, e.g. continuous delivery, we also integrate the suppliers and the entire organisation starts to concern itself with the new way of working.

That is one way of doing things – the path that best meets the exact needs of your company we decide upon together in line with your initial situation and your goals. We remain at your disposal during the entire changeover process.

Let's discuss your challenge!
Mutual trust is the essential prerequisite for such an important step as the changeover to an agile company. We would be more than glad to visit you for a non-binding consultation, to show you who we are, how we work and what we can do for you. We look forward to it!

borisgloger consulting GmbH
Lichtentaler Straße 7
D-76530 Baden-Baden
T +49 (0) 72 21.398 737-0
F +49 (0) 72 21.398 737-10
office@borisgloger.com
www.borisgloger.com

About borisgloger consulting GmbH
The borisgloger consulting GmbH with its headquarters in Baden-Baden and Vienna ranks in the DACH (Germany, Austria and Switzerland) region as one of the leading management consultants in the field of agile change management and agile product development. We focus on the management framework Scrum. The founder and Managing Director, Boris Gloger, is the first certified Scrum trainer worldwide and has trained over 5000 managers or rather project teams in this iterative approach model.
borisgloger consulting offers training and consulting for specialists and managers in the field of agile management. Founded in 2008, the borisgloger consulting GmbH currently has 20 employees and counts among its customers the Scout Group, otto.de, Deutsche Post and the Ergo Direkt insurance company. More information at www.borisgloger.com