



TOWARDS AGILE PRODUCT DEVELOPMENT - THE ROLE OF PROTOTYPING

Böhmer, Annette Isabel; Hostettler, Rafael; Richter, Christoph; Lindemann, Udo; Conradt, Jörg; Knoll, Alois

Technical University of Munich, Germany

Abstract

The complexity in all products and processes increases and the need for operative agility is seen as a key factor of success (Link and Lewirck, 2014). Agile approaches are extremely beneficial for situations with high uncertainty and a continuously changing environment. During the exploration new insights are gathered and an abstract vision becomes a concrete product idea. Based on testing and user feedback the feature set is adapted. The guiding principles are the interdisciplinarity and the strong design orientation. Rapid implementation of ideas into objects ("manifestation") triggers internal and external communication and feedback mechanisms. The prototype is seen as a process- and phase-spanning driver of the innovation process. However to date, there is no clear understanding of how an agile development actually looks like in the field of mechatronics and what the role of prototyping is in this context. Thus, the goal of this explorative study is to evaluate how prototyping is related to a mechatronic project path with regards to agile development. To this end, a prototyping roadmap is used to derive an agile mechatronic product development.

Keywords: Case study, Decision making, Early design phases, Innovation, Mechatronics

Contact:

Annette Isabel Böhmer
Technical University of Munich
Product Development
Germany
boehmer@pe.mw.tum.de

Please cite this paper as:

Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Innovation is a poorly understood and highly complex process (Link, 2014). Especially at its early stages innovation process are characterized and impeded by a high uncertainty about the problem and solution space. In these early stages, agile approaches such as design thinking are recommended: Their focus on customer and user needs helps to iteratively step processes into an increasingly concrete form (Link and Lewirck, 2014). According to (Hallgrímsson, 2012), prototyping is part of an iterative learning process to reduce uncertainties as early as possible. By creating numerous prototypes and testing with customers and users, new insights regarding the problem as well as solution space are gained. Prototypes can take many different forms: from very simple paper or cardboard models, mock-ups, function patterns to fully functional elaborations (Grots and Pratschke, 2009).

In software development, Scrum is the most widely used agile approach. The goal is to have a functional product at any time. The starting point of the development is a product vision, which roughly describes the requirements of the customers (Gloger and Margetich, 2014). A simple product with minimal functionality is advanced step by step and augmented with new features. These product increments are measurable intermediate results that are evaluated according to a certain (user) criteria. With every iteration, the product becomes more concrete. At the same time, however, the project becomes more immobile, since after each iteration, the team's range of options has decreased along with the depth and breadth of its decision tree. The target area of the planned solution usually changes due to gained knowledge. The obfuscation and fuzziness decreases as the project progresses (Link, 2014). A typically agile project path is illustrated in Figure 1 below.

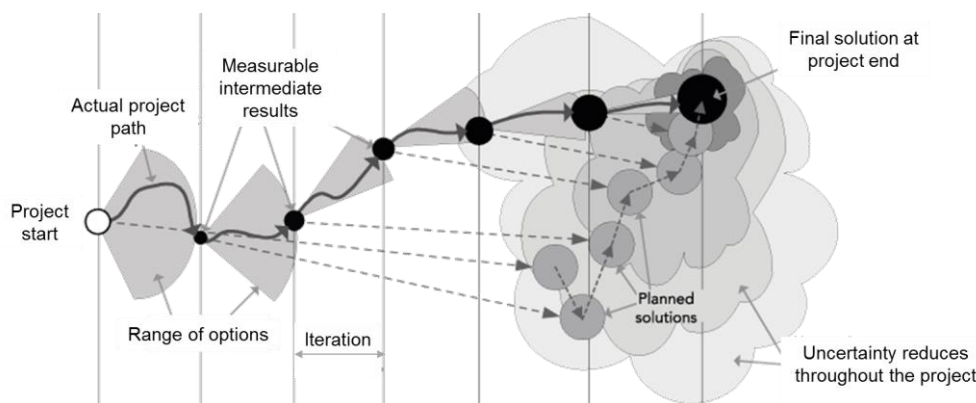


Figure 1. Agile project path according to (Oestereich and Weiss, 2008).

Up to now, the use of agile procedures is still limited to IT-projects, due to a lack of profound knowledge in transferring agile techniques into interdisciplinary projects (Klein, 2016).

According to Punkka (2012) in hardware development, up-front prototyping is used to experiment with something, to learn. Small, focused, experiments on partial solutions allow learning early, and efficiently. Up-front prototyping is different from traditional prototyping and shifts the objective of prototyping from validating to experimenting (Punkka, 2012). Likewise, innovation research states, that a pure phase consideration of an innovation process is no longer appropriate. The approaches shift from purely linear and sequentially to iterative and collaborative (Vetter, 2011).

The aim of this study is to identify an agile development path for mechatronic projects, by analysing the prototyping roadmap of interdisciplinary student teams. Therefore, this contribution presents a report from a 14 day Think.Make.Start. (TMS) Master course at Technical University of Munich (TUM), studying the role of prototyping in a "Agile Development" setting. The research work observes 40 single groups, adopting different strategies to address a design challenge, where they apply both traditional and agile approaches. The overall goal is to derive an object-driven development approach, by the systematic integration of agile approaches into the development process of mechatronic products.

2 BACKGROUND

The contemporary challenge of innovation in mechatronic products is the very short life cycle of today's technology. In order to successfully tackle this challenge, it is essential to get technology to market

rapidly. However, changes in product development processes can occur for many reasons, some of which are difficult to predict or anticipate (Thomke and Reinertsen, 1998). They are caused externally (e.g. change in customer needs) or internally (e.g. product complexity). The increasing rate of change and the requirement to be fast at the same time, leads to increased uncertainty about the product's definition. This leads to a limited effectiveness of traditional (product-) developments methods and thus requires the incorporation of new methods. One of the main strategies to handle this uncertainty is to accomplish the challenge of being agile that goes hand in hand with ever-present change (Link, 2014). According to (Punkka, 2012) up-front prototyping brings hardware development closer to the agile software development. Furthermore, the re-synchronization of planning at different levels, from operational and execution to strategic planning is simplified (Punka, 2012).

2.1 The role of Prototyping

The most known prototyping model within software development was introduced by Barry W. Boehm in 1988 as the spiral model. Its main objective is reducing uncertainties as early as possible by iteratively evaluating prototypes (Eigner et al., 2014). According to Budde (1992) there are four main aspects of prototyping: types of prototypes, objective of prototype, horizontal and vertical prototypes, as well as relationship between prototype and application system. Möller (2011) differentiates four types of prototypes: concept models, geometrical prototypes, functional prototypes and technical prototypes. (Hallgrimsson, 2012) classifies the prototype's objectives into four main categories: exploration, communication, usability and design, and verification (Hallgrimsson, 2012).

Horizontal prototypes represent a specific feature of the system, e.g. the human-computer interface without fully implementing them. Vertical prototypes focus on implementing a small set of features in a nearly-complete fashion. Likewise, (Thomke, 2008) differentiates between low and high fidelity prototypes. Low fidelity prototypes are inexpensive and can be produced rapidly for 'quick and dirty' feedback in the early concept phase of product development (Thomke 2008). Higher fidelity prototypes become increasingly important, to understand how close to a solution the effort is and to minimize the risk of failures (Thomke, 2008).

Prototypes are the only tangible object in the initial phase of an innovation process and thus provide a basis for discussion and ideation (Thomke, 2008). Punkka (2012) calls the costs for prototyping a kind of insurance to minimize the risk towards the end of the development project, which is traditionally relatively high. With many prototypes, especially at the beginning of the development process, assumptions and requirements are verified early, which can have a significant positive impact on the overall cost and duration of the project (Punkka, 2012).

Bertsche and Bullinger (2007) state that prototypes are not confined to specific phases of the product development process, but provide answers to questions as they arise. They are necessary to run experiments and they do not represent reality completely (Thomke, 2008). Rather they are part of an iterative learning process and an important tool for the discovery, evaluation and development of new product ideas (Hallgrimsson, 2012; Spinuzzi, 2002). These agile prototypes can be reduced to their essentials to gain knowledge (Smith, 2007). Simple build-ups enable faster testing of solution paths with the goal of rapid failure and early gain of knowledge (Kampker et al., 2016). Therefore, prototypes represent both, the knowledge and findings in a physical or virtual model (Gürtler and Lindemann, 2016).

According to Brown (Brown, 2009), prototypes test feasibility (technology perspective), viability (business perspective) and desirability aspects (customer perspective).

In summary, the use of prototypes within traditional and agile approaches varies significantly. Traditional approaches are plan-driven and a prototype represents the pre-defined product. In contrast, agile approaches make use of various prototypes to specify and develop a product in several cycles (Thomke, 2008). Uncertainty is reduced iteratively and changes that may occur (e.g. regulatory adjustments or new technologies) are considered along the way. Moreover, the development goal is continuously adjusted with respect to knowledge or (user) feedback gained. Processes, rules and working methods are not pre-determined, but are developed as the project progresses (Boehm and Turner, 2006).

2.2 Agility within Engineering

The fundamental assumptions differ strongly between agile and traditional approaches. While extensive planning is the basis for traditional approaches, agile approaches are hallmarked by prototypes and

artefacts (Smith, 2007). They assume that systems or products cannot be specified in detail or a priori. Instead, many iterations are set to gradually concretize and adjust the requirements and reduce uncertainty (Link, 2014).

In addition, the complexity of handling requirements for mechatronic products in product design has increased due to the integration of mechanical, electrotechnical and software components (Hehenberger and Bradley, 2016). In order to master such a development, agile focuses on parallel engineering with regards to systems engineering (Haberfellner et al., 2015). The guiding principles of agile product development are interdisciplinarity and strong design orientation (Vetter, 2011). Rapid implementation of ideas into objects ("manifestation") triggers internal and external communication and feedback mechanisms. The prototype is a process- and phase-spanning driver of the innovation process (Vetter, 2011).

However, for hardware development there are some constraints restricting the direct application of agile software frameworks (Buchholtz et al., 2011), e.g. the time to build prototypes. Such constraints of physicality are facilitated by the presence of Makerspaces. These high-tech workshops boost the innovation capability of its community by providing a space where innovative people meet, socialize and collaborate on new ideas or do-it-yourself projects. They are part of the new paradigm of Open Innovation Ecosystems (OIE) and innovation fostering environments. They base on principles of collaboration, co-created value, exponential technologies and a culture of exceptionally fast acceptance (Darrin and Krill, 2016).

The rate of innovation is a function of the number of people connected and exchanging ideas. In this context, Makeathons are a fast way to explore ideas in an interdisciplinary team with high technical and market uncertainties. Focussing on iterative prototyping the main feature set is discovered in a minimum of time (Komssi et al., 2015). The exchange of knowledge and the creative collaboration enables not only a faster but also more flexible product development processes.

Such experimental approaches are test driven by nature and foster change as a chance for flexibility. They follow the underlying idea of agility, which is similar to the Plan-Do-Check-Act (PDCA) cycle of Deming (1992). However, there is no clear understanding of how an agile development actually looks like in the field of mechatronics. Both, the product itself as well as the correlated process evolve over time with the project's progress. The use of methods ("how to") is well-understood for product development, however, the overall approach ("what") is unclear. When not specifying products in advance, agile prototypes are necessary to facilitate "learning by doing" (Kampker et al., 2016).

3 RESEARCH METHODOLOGY

The aim of this paper is to study is to identify an agile development path for mechatronic projects, by analysing the prototyping roadmap of an interdisciplinary engineering class. A key motivation of agile approaches is to build artefacts, measure their impact when introduced to the environment and learn or rather evaluate the interaction with them (Dennehy et al. 2016).

However, it is difficult to delineate the term evaluation due to its complexity and variety of application areas. Taking into account not only on viability, but also attributes like feasibility and desirability challenges the objectivity, comparability and traceability of any evaluation. As a result, this work deals with "little" and "partially known phenomenon" and therefore makes use of an explorative research paradigm. It allows for flexibility in the search for data as well as openness and creativity during the research process. Both qualitative and quantitative data are collected during this exploratory research (Shaffir & Stebbins, 1991).

By combining the qualitative and quantitative results, a symbiosis takes place to establish a first generic concept (Cash et al., 2016). This corresponds to the qualitative part of the "partially known phenomenon". Through the inductive creation of an agile framework and a logic for structuring the components, a qualitative basis is created, on which finally a quantitative comparison of the collected data from TMS can take place.

The research work observes 40 single groups during a practical course (TMS) at TUM, adopting different strategies to address a design challenge, where they apply both traditional and agile approaches. The overall goal is to derive an object-driven development approach, by the systematic integration of agile approaches into the development process of mechatronic products.

The application of the methods is free and there are no strict categories or criteria on when to apply more traditional or agile methods. The studied development paths are compared to general expectations of

what the case would have looked like had the teams not been briefed and to other events casually observed.

Guided by a literature research, the study focuses on the prototyping roadmap (e.g. generated artefacts, product increments and their correlation) and the object-driven development as our research variables. Figure 2 illustrates the theoretical model to explore the presumed relationship. The mediation effect on the capability of implementation and the logic linkage are discussed. Likewise, the moderation effect of a knowledge base on the implementation capability and the scope for decision making on the logic linkage are elaborated. The type of data used to represent and measure the variables in this research are outlined in the following.

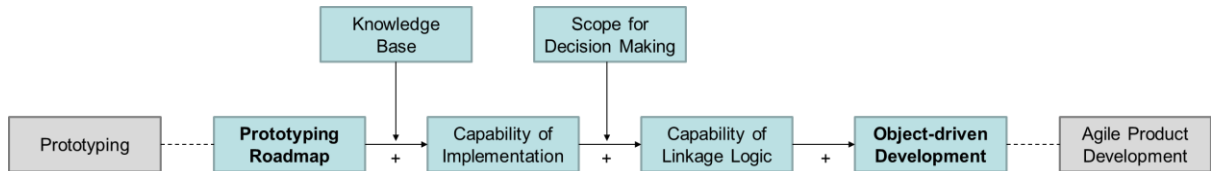


Figure 2. Theoretical model "What?", "How?", and "When" of effect of prototyping roadmap on object-driven development according to (Cash et al., 2016)

3.1 Setting of Think.Make.Start.

Think.Make.Start. (TMS) is a practical course at the TUM in cooperation with UnternehmerTUM. Each semester ~50 master students take part and develop their ideas to create a potential business around them within just 14 days. TMS brings together students from different backgrounds, such as Mechanical Engineering, Informatics, Computer and Electrical Engineering, School of Management as well as others (Medicine, Communication Management, etc.). The students allocate themselves into teams under the constraint that each team must represent at least three different faculties.

Since 2015, 189 students participated and 38 innovation projects emerged from that course. The projects' topic is freely chosen, but limited to a budget of ~400 EUR. The teams are supported by coaches of the corresponding faculties and have free access to the MakerSpace. TMS is characterized by time pressure, competition and an open community. The students learn agile and traditional methods and principles, but the team- or time-specific application is not predefined. The resulting agile product development approach is inspired by integrating knowledge and methods from different disciplines, using a real synthesis of approaches. The research focuses on the application of elements or rather methods out of the following agile frameworks: Makeathon, Scrum, Design Thinking as well as Lean Startup. The central aspect of the agile product development is the Munich Procedural Model (MPM) by Lindemann (2007) (see Figure 3).

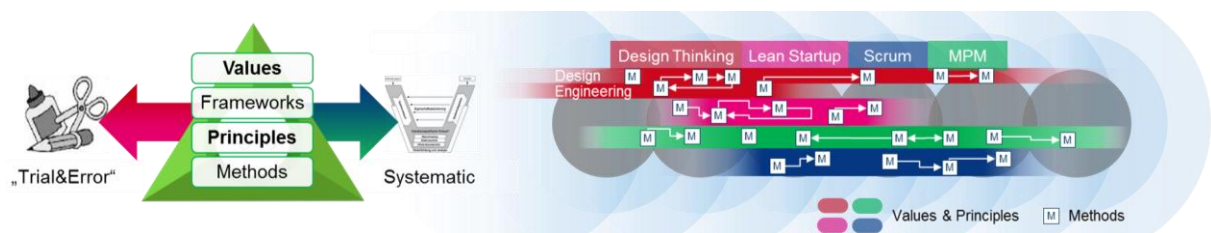


Figure 3. Concept of values, principles, as well as frameworks and methods.

3.2 Data Collection

To identify an agile development path for mechatronic projects, the prototyping roadmap of each interdisciplinary team was analysed. Data was collected throughout the course and repeatedly from the same students along the lines of a longitudinal study. In order to obtain objectivity, the data is acquired using six different approaches and focuses on four different aspects.

The main aspect of an agile project path is the "prototyping roadmap", which was complemented by the "daily progress presentation", a list of "generated artefacts", and a database illustrating the "overall project path" (see Figure 4). The prototyping roadmap was gathered by providing the students an online "TMS Pad", where they uploaded all of their prototypes. User stories are formulated in order to implement these with a prototype. The underlying hypotheses were verified or falsified using predefined

success criteria. To plan the next step, each prototyping box included learnings obtained with each prototype testing.

For transparent data acquisition, each team had an additional online database available for project work. The data was complemented by a daily questionnaire as well as observations and interviews of two independent students to deepen the understanding.

A central aspect of TMS are the daily progress presentations where the students present their current status (e.g. covering their achieved goals, new insights, and lessons learned) to acquire feedback (e.g. on generated artefacts). This allows a continuous (re)planning based on the insights and knowledge gained and results in an iterative product development.



Figure 4. Used methods for data collection and related analysis aspects.

The teams completed TMS with a 500-word report, reflecting on the last two weeks. Thus, an overall retrospective of the entire project path was obtained by each team.

4 FINDINGS

Most teams were quickly applying agile principles, whereas some also followed strictly the known traditional methods. Applying agile procedures, a few teams struggled with the chaotic aftermath of wrongly applied agility. Meaning that, some teams had challenges capturing the big picture while not getting lost within testing and incremental development. This often implied the disappearance of familiar mechanisms of planning and control as well as the apparently non-systematic form of documentation. A structured (daily) documentation seems crucial for agile approaches. It facilitates to focus on the most relevant aspects of the project and enables reviews for certain decisions made.

A few teams did not document their predefined hypothesis and the decision made after each testing. Therefore, their prototyping progress did not follow a strict plan but rather an unsystematic exploration. A plan-driven development process compensates being not aware of every detail all the time. In contrast to this "just following the plan" mentality, object-driven processes have the necessity of constant replanning and adapting to the knowledge gained. Based on the project idea, the vision was the overall goal to achieve. Although it was not specified in detail and may have been abstract. Breaking down the vision enabled the specification according to the user needs and technology change. This approach seems extremely appealing when having high uncertainty or rather a gap of knowledge.

4.1 Prototyping Roadmap

Analysing the different teams and projects, it is noticeable that there are two main strategies of agile product development. On the one hand, teams consider their vision as a black box; trying to break it down by gathering information and exploring different versions of prototypes. Step by step the teams build upon the previous versions and add or remove features. Most of the time, the product is built several times using different types of manufacturing or by updating the old version. Doing so, the team reached a high level of integral structure and functionality. Usually, these high-fidelity prototypes had a low level of complexity, meaning less components and simple component interconnections.

On the other hand, teams were building more complex products. An early modularization facilitated the development. The teams focused on the minimum feature set creating value, while putting non-critical features on the shelf. The implementation of the minimal feature set then was critical for the success of the overall project idea. These modules were manufactured in detail while the rest of the product stayed at a concept stage. These projects gained a lot from disassembling existing products, in order to present the overall product idea at a very early stage. The missing functionality of the other features were imitated or faked for user tests.

Based on the data collected, every development process could be separated into a vertical or horizontal strategy (see Figure 5). A vertical prototype focuses on a certain feature and increases the level of functionality or rather detail (paper prototype, Arduino prototype, etc.). They seemed most appropriate when a certain complex feature of a system is poorly-understood and needs to be explored (e.g. proof-of-concept). It addresses the question "how" to implement a certain feature.

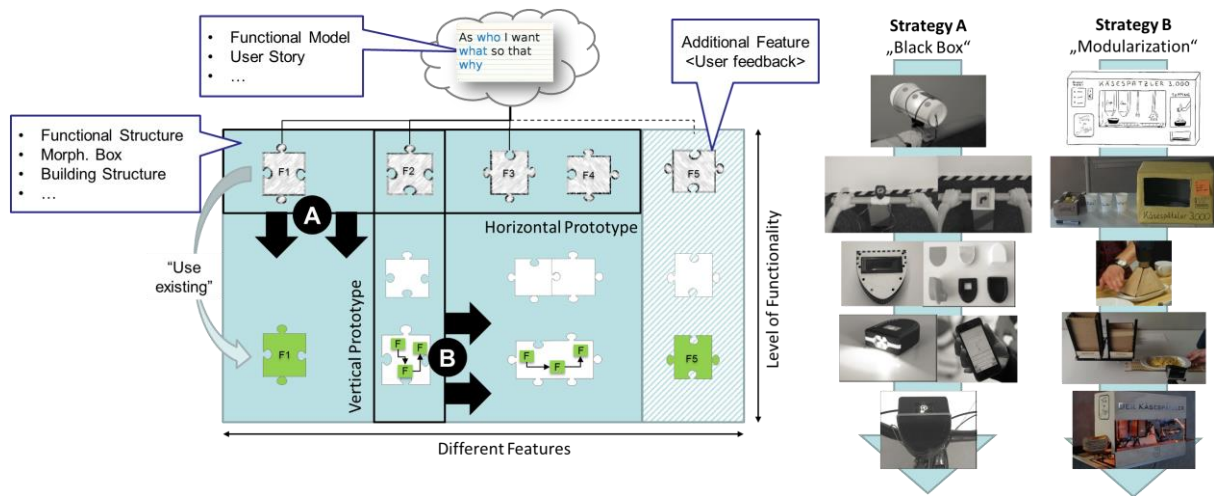


Figure 5. Horizontal and vertical prototypes as part of an agile product development.

A horizontal prototype explores different feature sets and ways of implementation (material, design, etc.). They are appropriate for understanding relationships across a broad system and address the question "what" to implement with regards to the user needs and feedback. The methods used did not define the process but support the generation of prototypes.

According to the user feedback as well as the team competency (experience and knowledge) the product was specified iteratively. Uncertainty decreased over time and aspects regarding feasibility, viability, and desirability were explored towards a successful innovation. These three aspects of the final product interacted with each other and made it necessary to jump between horizontal and vertical approaches in order to get a well-defined solution.

4.2 Object-driven Development

Prototyping can be seen as central method within the innovation process and is supported by the Makerspace and its community. For agile product development, the role of architecture is important, since extensive refactoring is potentially hazardous and thus not always feasible. This is facilitated by a high degree of modularization. However, during TMS product architectures were rarely stable.

Based on the information accessible and the teams' background, agile or traditional principles were more applicable. Thus, instead of performing a lean experiment, teams new to the field undertook a technology or market research. Having a small knowledge base may result in unnecessary tests. However, first insights gained by quick user tests were indisputably useful for the projects.

Each iteration of the object-driven process had experimental and explorative characteristics. Predefined hypotheses were tested by a prototype. Therefore, a certain objective (e.g. communication) was defined for the prototype. Specific characteristics (e.g. wood material) and types of prototype (e.g. Design Prototype) were used to measure data and thus evaluated the assumptions made. Based on the lessons learned the objective may have shifted due to new insights or gains in knowledge. Consequently, the project ideas shifted within the initial problem and solution space. This prototyping logic, illustrated in Figure 6, allowed a result-driven (re)planning of the development.

A prototype represented both, the then current knowledge and findings in a physical or virtual model. Especially for multi-disciplinary teams a prototype made the project status clearer and supports the internal team communication. Prototypes were also continuously representing the product status incorporating every discipline. Different types of prototypes were used to focus on discipline specific aspects (e.g. Design vs. Functional Prototype). However, they also led to a common vision and improved the understanding of each component and accompanied constraints (e.g. geometric dimensions).

For all TMS projects, the generated artefacts (e.g. mind maps, functional models) were assigned to one of the three categories: feasibility, desirability, and viability. However, it was found that there are also artefacts that are correlated to both feasibility and viability (e.g. benchmark) or rather desirability and feasibility (e.g. CAD-Mockup). Nevertheless, the artefact categorization supported the teams project progression. It was easy to adjust the next steps by incorporating the generated artefacts.

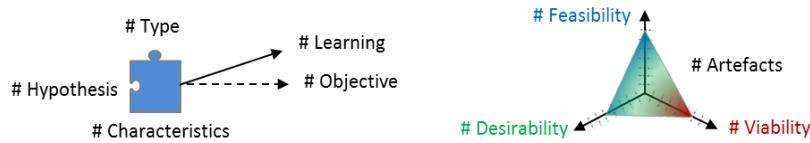


Figure 6. Linkage logic of product increment as well as artefact categorization.

In summary, the linkage logic allowed for an object-driven project path, that was not predefined and enabled the ability to constantly adapt to a changing environment and new insights.

4.3 Agile Product Development

It is required to combine both principles "technology push" and "demand pull" for a successful innovation (Vetter, 2011). Transforming customer needs into product requirements asks for a joint understanding of the complete situation (problem- and solution space) within the team (Link and Lewirck, 2014). In TMS this is supported by visualization of the ideas and a moderated process. The product development was mainly characterized by a prototype roadmap. Starting with an abstract vision, the prototypes became more concrete with each iteration. As the resolution of the prototype increased the range of options was decreasing and the initial project idea shifted to adapt to the gained knowledge.

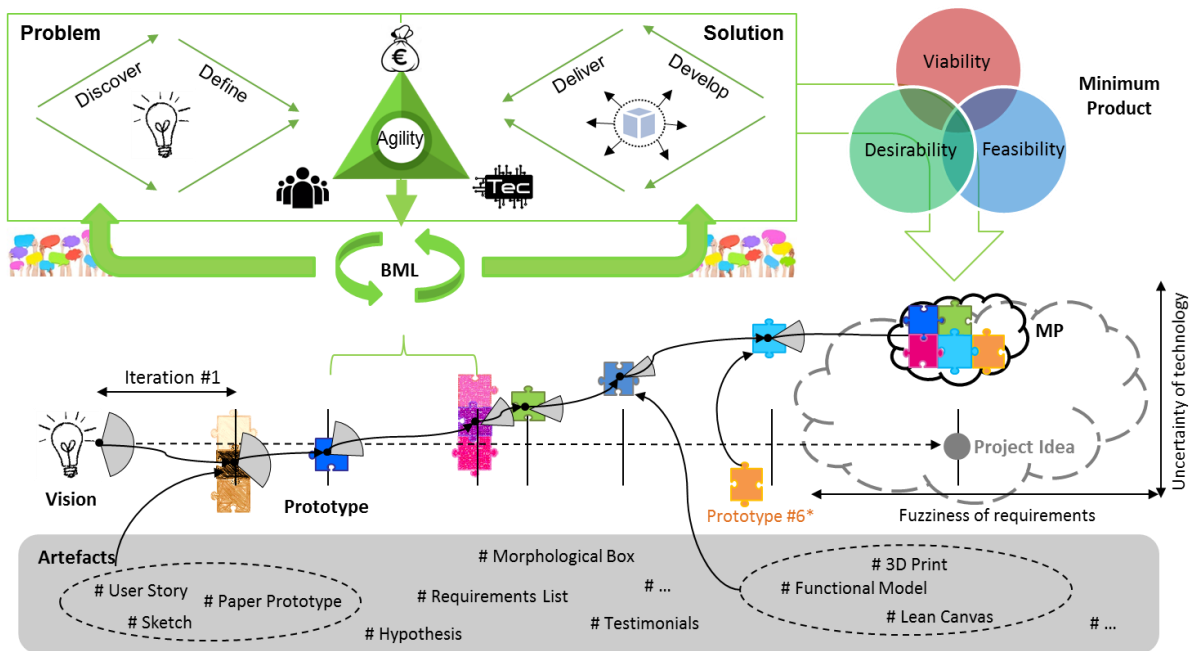


Figure 7. Agile Product Development Model.

The purpose for agile principles is to foster speed and flexibility, starting with an abstract idea and finishing with a detailed product. Methods support the creation of certain artefacts needed for the prototyping development. With each iteration, the team gained new insight into various aspects of the project. The lessons learned lead to deviations from the initial project idea, indicated by the vertical offset of the increments. Being open to feedback and change, was often accompanied by adding new features later in the process, indicated by prototype #6* in Figure 7. Subsequently, the range of options increased once again, which is why change is seen as an opportunity for flexibility.

First teams focused on building the right product (incorporating FDV) before building it right. As the projects progressed the need for traditional methods increased. Shifting from agile to more traditional principles. Considering product architecture and engineering change management. While prototyping was generally used, the transition to such a well-documented product development is a challenge.

Transparency is one of the most important values for agile projects. Having relevant information available, the teams were able to make decisions and replan the project quickly the moment it changed or new insights occurred.

A clear order regarding the two prototyping strategies was observed (see Figure 8). The use of prototypes changes during the development from horizontal prototypes to more vertical prototypes. Even in later iterations, horizontal prototypes were still used, based on new insights (e.g. reconsider selected designs, find new solution possibilities). Both approaches were needed in order to explore the complete solution space. However, best results were obtained by switching perspectives frequently.

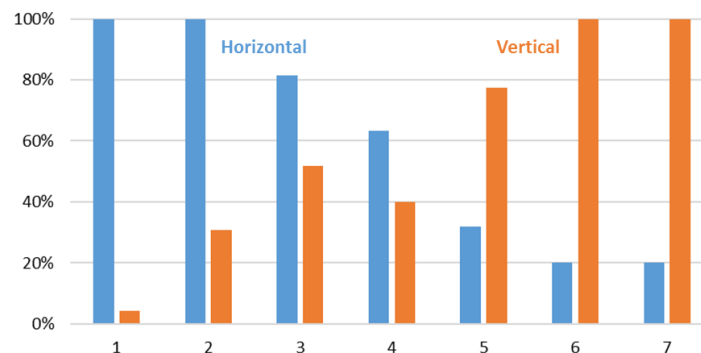


Figure 8. Horizontal ("what") and vertical ("how") prototypes for each iteration for all teams.

5 DISCUSSION AND OUTLOOK

In this work, a group of students was observed as they were going through an agile development cycle for mechatronic products. Their work was analysed to understand the influence of prototyping on agile development. To that end, the prototypes were classified in horizontal and vertical prototypes. It could be observed that strategies iterating between the two types of prototypes lead to well defined projects and the use of prototypes shifted from horizontal to vertical as the projects progressed.

Agile principles support the horizontal prototyping strategy, whereas traditional ones facilitate the vertical implementation of a feature. With an increasing level of fidelity more traditional principles were needed in order to achieve a well-developed prototype. Focussing on one prototyping strategy, teams got stuck within the process, not knowing what to do next. There, either agile coaches helped by challenging the team with the user problem and the necessary minimum feature set to solve that. Or the coaches helped with different ways on how to implement a feature to overcome mental barriers.

For teams, who used traditional methods not necessary thoroughly, but early in their project, the transition to a more detailed product development was smoother. In contrast, teams that just switched from agile to traditional approaches struggled to proceed with the project due to massive planning.

Retrospectives prevented teams blindly following a process. With each reflection, the objective and the needed characteristics of the next prototype were discussed. Doubtful team members hindered agile strategies and lead to stagnation of project progress at a certain level.

Teams mostly following traditional approaches were mostly optimizing the initial idea. Compared to competing teams, there was no breakthrough throughout the process. However, the team hold on to the initial idea and followed the plan. An established mechatronic development process cannot be adapted to agile procedures overnight. Instead, present conventional models should be supported by agile procedures in an appropriate way. Future research aims to deepen the understanding for object-driven development process to explore the solution space, supported by methods. Thus, merging traditional and agile frameworks, in order to benefit of these two approaches in the best way.

While the products in question were at times very complex, sometimes even combinations of several mechatronic systems, only a very limited scope can be covered with prototypes due to the limited time-frame. The role of the prototypes in more complex systems is thus harder to gauge. However, as seen in Figure 8, the use of horizontal prototypes diminishes over time and the use of traditional methods is increasing. Furthermore, as the focus lies on prototyping the most uncertain and most critical features first, it can be argued that there is some applicability of the findings to very complex products. But especially when the complexity lies in the integration of modules rather than building components, further research is required to identify the role of prototypes in these projects.

Finally, the studies focus on just the first two weeks of development, thus there is no data on the long-term effect of the application of these prototypes. While the fact that out of the 38 teams, 7 currently still exist as start-up companies hints at a certain long-term validity of the approach, further research is required to establish a good understanding.

REFERENCES

- Bertsche, B. and Bullinger, H.-J. (2007), *Entwicklung und Erprobung innovativer Produkte - Rapid Prototyping: Grundlagen, Rahmenbedingungen und Realisierung*, VDI-Buch, Springer, Berlin, Heidelberg.
- Boehm, B. and Turner, R. (2006), *Balancing Agility and Discipline: A guide for the Perplexed*, Addison-Wesley, Boston.
- Böhmer, A., Beckmann, A. and Lindemann, U. (2015), "Open Innovation Ecosystem - Makerspaces within an Agile Innovation Process", *ISPIIM Innovation Symposium*, Vol. 2015, pp. 1–11.
- Brown, T. (2009), *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*. Harper Business.
- Buchholtz, G., Buckow, J., Denger, C., Reuner, T., Lamdgraf, K., Rüttinger, Anton, Schütz, Oliver and Weilkens, T. (2011), "Agiles Projektmanagement für Systeme im regulatorischen Umfeld", *Tag des Systems Engineering 2011*, Vol. 2011, pp. 161–172.
- Cash, P., Stanković, T. and Štorga, M. (2016), "Experimental Design Research: Approaches, Perspectives, Applications", *Springer International Publishing*, Cham.
- Darrin, M.A.G. and Krill, J.A. (2016), *Infusing Innovation Into Organizations: A Systems Engineering Approach*, CRC Press, Portland.
- Eigner, M., Roubanov, D. and Zafirov, R. (2014), *Modellbasierte virtuelle Produktentwicklung*, Springer Berlin Heidelberg; Springer Vieweg, Berlin, Heidelberg.
- Gloger, B. and Margetich, J. (2014), *Das Scrum-Prinzip: Agile Organisationen aufbauen und gestalten*, Schaffer-Poeschel Verlag für Wirtschaft Steuern Recht GmbH, Stuttgart.
- Gürtler, M. R. and Lindemann, U. (2016), "Innovationsmanagement," in *Handbuch Produktentwicklung*, München: Hanser, pp. 483–512.
- Haberfellner, R., Weck, O.d., Fricke, E. and Vössner, S. (2015), *Systems Engineering: Grundlagen und Anwendung*, 13th ed., Orell Füssli, Zürich.
- Hallgrímsson, B. (2012), *Prototyping and Modelmaking for Product Design*. Laurence King Publishing.
- Hehenberger, P. and Bradley, D. (Eds.) (2016), *Mechatronic Futures*, Springer International Publishing, Cham.
- Kampker, A., Förstmann, R., Ordnung, M. and Haunreiter, A. (2016), "Prototyping in the Agile Development Management", *ATZ worldwide*, Vol. 118 No. 7-8, pp. 66–70.
- Komssi, M., Pichlis, D., Raatikainen, M., Kindstrom, K. and Jarvinen, J. (2015), "What are Hackathons for?", *IEEE Software*, Vol. 32 No. 5, pp. 60–67.
- Link, P. (2014), "Agile Methoden im Produkt-Lifecycle-Prozess – Mit agilen Methoden die Komplexität im Innovationsprozess handhaben, in Schoeneberg, K.-P." (Ed.), *Komplexitätsmanagement in Unternehmen*, Springer Fachmedien, Wiesbaden, pp. 65–92.
- Link, P. and Lewirck, M. (2014), "Agile Methods in a new area of innovation management", *Science-To-Business Marking Conference*.
- Möller, E. (2011), *Handbuch Konstruktionswerkstoffe - Auswahl, Anwendung, Eigenschaften*. München: Hanser.
- Oestereich, B. and Weiss, C. (2008), "Agiles Projektmanagement: erfolgreiches timeboxing für IT-Projekte", 1. Aufl., Dpunkt-Verl., Heidelberg.
- Punkka, T. (2012), "Agile Hardware and Co-Design", *Embedded Systems Conference*, No. ESC-3008.
- Smith, P.G. (2007), *Flexible Product Development: Building Agility for Changing Markets*, 1st ed., Jossey-Bass, San Francisco.
- Spinuzzi, C. (2002) "A Scandinavian Challenge, a US Response: Methodological Assumption in Scandinavian and US Prototyping Approaches," in *Proceedings of the 20th Annual ICCD*, pp. 208–215.
- Thomke, S. (2008), "Learning by experimentation: Prototyping and testing", in Loch, C.H. and Kavadias, S. (Eds.), *Handbook of new product development management*, Elsevier, Amsterdam, pp. 401–420.
- Thomke, S. and Reinertsen, D. (1998), "Agile Product Development. Managing Development Flexibility in Uncertain Environments", *California Management Review*, Vol. 41 No. 1, pp. 8–30.
- Vetter, M. (2011), "Praktiken des Prototyping im Innovationsprozess von Start-up-Unternehmen", *Gabler Verlag*, Wiesbaden

ACKNOWLEDGMENTS

Great acknowledgments to the team of UnternehmerTUM and students of TUM for their support and enthusiasm. Besides Technical University of Munich, the Vector foundation, and the Zeidler research foundation funded TMS. In conclusion, a big thank you to the MakerSpace team for their support.