Towards a Framework for Agile Development of Physical Products

Influence of Artifacts and Methods

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

Maximilian Meinzinger
Institute of Product Development
Technical University of Munich
Munich, Germany

Rafael Hostettler
Institute of Robotics & Embedded Systems
Technical University of Munich
Munich, Germany
rh@gi.ai

Alois Knoll
Institute of Robotics & Embedded Systems
Technical University of Munich
Munich, Germany
knoll@in.tum.de

Udo Lindemann
Institute of Product Development
Technical University of Munich
Munich, Germany
lindemann@pe.mw.tum.de

***Abstract*— A typical agile project is characterized by fuzziness. However, this decreases from iteration to iteration by the learning effect of measurable partial results, like prototypes or product increments [6, p. 20]. In the agile process, artifacts are generated by the application of methods. They contain all the information that is required by the product development process and reflect the current state of knowledge. They are thus information carriers. These artifacts can be either physical / tangible or virtual / immaterial. The goal of this work is to evaluate the influence of methods and artifacts on agile projects to serve as a basis to derive an agile framework. To this end, the collected data is structured into an analysis framework allowing for a systematic evaluation. Therefore, the Makeathon Think.Make.Start. was analyzed. During this agile development of innovative products, artifacts are a central element and carrier of information.**

Keywords—agile; prototype; makeathon; iterative; framework

I. Introduction

Innovation is a hardly understood and highly complex system [1, p. 67]. Especially the early phases of innovation processes are characterized by a high uncertainty about the problem and solution space. Agile approaches such as design thinking are recommended in these early stages to mitigate the uncertainty. The focus on customer or user needs facilitates the iterative concretization of the problem-solution fit by empathizing with the user [2, p. 5]. These insights are gained by creating various prototypes that enable interactive user tests. Based on the test variable, prototypes are of different forms and types. They vary from very simple paper or cardboard models, mock-ups, function patterns to fully functional elaborations.

Using specific prototypes, the product becomes more concrete with every iteration. The fuzziness of the project decreases and the requirements become more specific. At the same time, however, the project becomes more immobile. With each iteration, the team's range of options decreases along with the depth and breadth of its decision tree. However, the initially planned solution usually changed in the process due to the gain of knowledge [1].

With Think. Make. Start. [3], a lecture format has been created that follows this iterative methodology for the development of de-novo physical products. Based on data acquired during the fourth instantiation of the format, insights on the role of artifacts created in the process as well as of the methods applied from the agile development corpus were gained. To systemize these insights, they are structured through an analysis framework.

II. Focus of the study

The focus of this work focuses on the early stages of product development. The product development process is central element of an innovation process and generally includes all basic activities from the identification of innovative ideas, the development of prototypes to the start of production [4, p. 488]. All activities, from the idea to the practical implementation in the form of market launch, are therefore considered. These are subject to the influence of various disciplines, e.g. organization, marketing or construction [5, p. 3]. They are therefore characterized by high complexity and a priori unknown interdependencies.

One way of counteracting this is to use agile methods throughout the product lifecycle [1, p. 65]. Such agile approaches are widespread, especially in software development [6, p. 18], while the development of physical products is traditionally oriented on traditional and linear approach models [1, p. 84]. However, there are now also approaches to integrating and applying agile methods [7, p 21]. For example, companies are increasingly trying to implement Scrum as a project management method to make their development activities faster and more efficient [8, p. 87].

In practice though, there are investigations which reveal problems and challenges in the transfer of agile methods to physical products. For example, their implementation is poor due to the high organizational effort in companies [9, p. 5]. The actual targets, e.g. shorter development times or lower costs are not achieved as a result [9, p. 2]. Another example is the limited applicability of agile methods due to the physical product design [10, p. 10]. This is illustrated by the example of the creation of a prototype, which can be done quickly and cheaply in software development because of the virtual nature, but not for a physical product.

III. Background

A. Artifacts within Agile Product Development

Artifacts are a central element of this work, thus their meaning in the context is further explored. The word "artifact" comes from the Latin and derives from arte = with skill and factum = the made. Depending on the context, different meanings are attributed to the term. For example, an artifact in archeology is an object which has been given its form by human action, whereas in the electronics a disturbance signal is meant [11].

In Unified Modeling Language (UML), an artifact is defined as an element that arises during a development process, during the deployment, or during the application of a software system. This is a physical piece of information, such as a source code file or a table within a database [12, p. 213]. [13, p. 12] defines artifacts beyond the UML as all the results of software development, which are represented in a particular notation (e.g., natural language or programming language). These may also be intermediates. Here, a distinction can be made between material (source texts, drafts, documentation, etc.) and immaterial artifacts (for example, methods, knowledge or concepts) [14, p. 81].

Apart from software development, [15, p. 21] describes an artifact as an object, on which knowledge is stored as information. It is interesting to note that all objects represent, in a certain form, an artifact. For example, even a simple product like a table can be an artifact because reverse engineering can generate information. [16, p. 49], in his definition, includes the origin and use of the artifact. It describes the term as a virtual or physical business object, which can be generated, processed or eliminated in processes.

While the definitions that have so far been used to define the term artifact are very broad, [17, p. 37] restricts the understanding in the context of machine and plant engineering. An artifact is a component of a procedural model. In this case, artifacts describe in document-like form, which components are to be developed within a project and which (intermediate) results are produced. They therefore include what is to be developed with respect to the product.

As illustrates in Fig 1, artifacts are created when certain activities are performed by employees who have a specific role [19, p. 75]. Existing artifacts can also be modified and the use of a given artifact can trigger the need for a new artifact. Influence on the process of production has the use of the method as well as information (e.g., guidelines) or framework conditions (e.g., presence of tools) [18, p. 443].

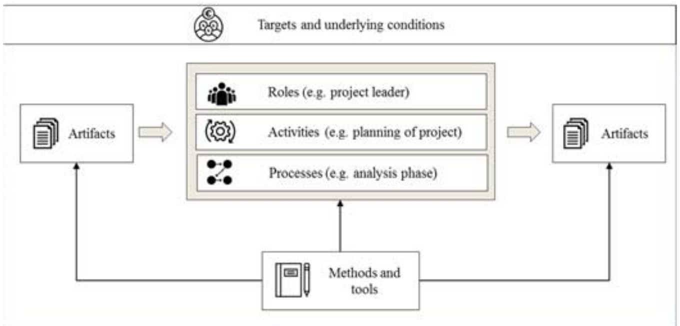

Fig. 1. Elements of an agile procedure model (according to [20, p. 17, 21, 18, p. 443].

Artifacts contain all the information that is required and required by the product development process. They can be either physical / tangible or virtual / immaterial. They are thus information carriers. Under the influence of methods and tools as well as guidelines and frameworks, artifacts are generated, edited or eliminated within a procedural model. Similarly, triggering the need for a new artifact by using an old one is possible. Depending on the respective situation, they flow into the different submodels (roles, activities, sequences).

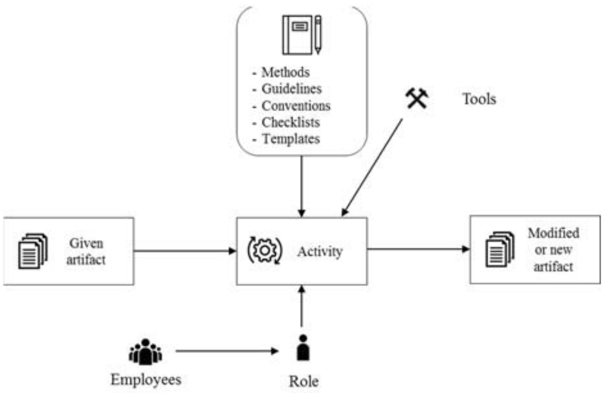

Fig. 2. Correlation of artifacts and activities (according to [18, S. 443]).

An action model defines who is responsible for the creation of certain elements (product model) at which time (run model). Likewise, it is specified how the sequence of the procedure is designed (activity model) [21]. Overall, an approach model thus presents "who has to do as what and when". Developer support is provided through the provision of methods and tools. We must always take account of the requirements and general conditions. These may be, for example, externally prescribed standards (e.g., ISO standards) as well as internal specifications (e.g., compliance policies). Incorporating the correlation of artifacts and activities the following correlation of elements within an agile procedure model can be derived (see Figure 2).

B. Agile Product Development Model

Linear approach models are often described as sequential, classic or traditional [22, p. 5, 23, p. 70]. A well-known model, often referred to as a typical example or synonym, is the waterfall model [24, p. 329]. These procedures are characterized by extensive and complete planning at the beginning of the project [22, p. 5].

Evolutionary or iterative approach models developed first. One of the first known approaches was the spiral model developed by Barry W. Boehm [25, p. 63]. In the iterative approach, the development proceeds from a general specification step by step into ever more concrete tasks [26, p. 75]. The entire system is thus developed in several iterations. Planning takes place here only in short-term time horizons, whereby adjustments can be made after each run through experience and learning effects [27, p. 21]. In analogy to linear models, the procedure here is top-down, but jumps and deviations from the standardized procedure are permitted (see Figure 3b) [18, p. 560]. In the iterative approach, the planning goes into the background and the result gains in importance. These results are related to the prototypes built [28, p. 55].

In the case of agile procedures, long-term planning is not the focus, the customer and the end result have the highest priority [28, p. 55]. Since the planning is not planned for long periods, requirements are initially only roughly defined [27, p. 23]. The detailing then takes place during the project. The main focus in the agile approach is the adaptability and flexibility, the unpredictability of certain events is consciously accepted [1, p. 74]. The entire development is divided into iterations within which the individual development phases are passed without a fixed sequence [6, p. 19]. Rather, it is the responsibility of the team [23, p. 47] to complete the required phases. As can be seen in Figure 3c, the individual steps can even be performed in parallel.

An important feature is the provision of prototypes or rather potentially deliverable product components, the so-called product increments [29, p. 36, 30, p. 242]. This step-by-step deployment implies that requirements in the project process are met in turn [28, p. 55]. A typical agile project is characterized by fuzziness. However, this decreases from iteration to iteration by the learning effect of measurable partial results, like prototypes or product increments [6, p. 20].

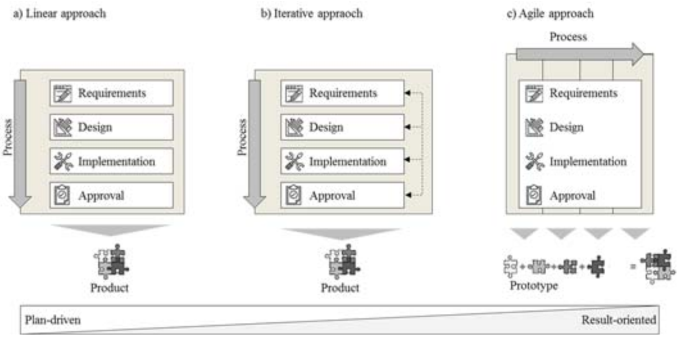

Fig. 3. Characteristics of procedure models (according to [23, p. 70, 17, p 47, 27, p. 19].

Furthermore, the procedure for all agile methods is strongly based on the implicit knowledge and skills of the team members [31, p. 7]. If the development and the advance of the project are not sufficiently documented, difficulties can be experienced at a later stage in individual team members [32, p. 333]. In this context, [33, p. 33] note that the implication of implicit

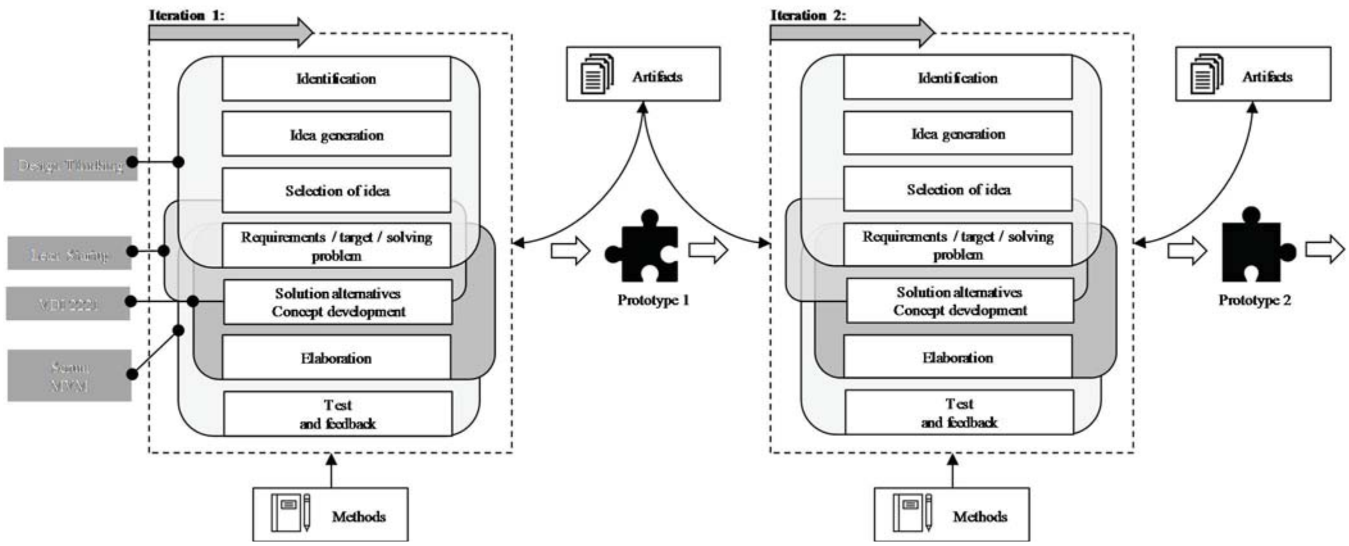

Fig. 4. Illustration of agile approach with incorporated artifacts and Prototypes.

knowledge can make a great contribution to the flexibility and speed of agile methods.

In conclusion, it is important to note that a certain degree of documentation is essential in the development of physical products. As a result, a large number of artifacts are created during the innovation process. Since this subject area is currently under investigation, the focus will be on artifacts in the following chapters.

C. Classification of Artifacts

A first approach to classify artifacts is to divide them by individual project phases. [34, p. 8] divide artifacts of agile project management in software development into the following categories: requirements, program code, tests, delivery items, planning and control. These can be divided more generally into planning, requirements and specification, change management, and testing [35, p. 3] Based on this, [36, p. 26] developed a detailed categorization of artifacts:

- Planning (e.g., Sprint Plan)
- Requirements (e.g., Product Backlog)
- Development (e.g., software codes)
- Tests (e.g., test reports)
- Change-Management
- Governance artifacts

The last-mentioned category contains all artifacts that cannot be classified in any of the previous categories. These are, for example, risk assessments or product standards.

Furthermore, an outline can be created to determine what kind of artifact it is. A division of requirement artifacts can be done into three categories according to [37]: Container, individual element or solution element.

A container can be defined as a sort of collection point. These are documents that keep the project together. This can either be a so-called artifact container, which consists of individual or solution elements. An example would be the Product Backlog. Furthermore, a container may also be a generic document that is continuously updated (e.g., a text document for describing the product) [37, p. 136]. Individual elements are either user-oriented (e.g. use-case) or technical (e.g., system requirements). The last kind of artifact is the solution element. These are either concrete (e.g., GUI mockup) or abstract (e.g., a spreadsheet) [37, p. 137].

A further classification of requirements artifacts was made by analyzing different strategies in the implementation of the requirement engineering. Three different directions of impact could be determined, into which the resulting artifacts can also be divided. It is a matter of …

- solution orientation: focus on the customer
- functional orientation: focus on applications and interfaces
- problem-orientation: focus on business and economic needs [38, p. 19]

According to [39] the data collected is classified to one of the three categories: feasibility, viability and desirability. Feasibility measures the design's technical functionality, desirability its value for the customer as well as the likelihood of purchase, and viability the ability of the designs to fit into time- and budget constraints. Using these three variables, a three-dimensional space is created

IV. Research Design

A. Think.Make.Start. – An Agile Framework

Think.Make.Start. (TMS) is a practical course at the TUM in cooperation with UnternehmerTUM. TMS brings together 50 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.

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, a large workshop. 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.

B. Research Question

In order to formulate the components of a best practice framework, it is necessary to understand the role of artifacts and methods on an agile process. Thus, the following research questions or hypotheses are the main focus:

- When applying agile procedures in physical product development, methods are used and artifacts are created.
- There are dependencies and interdependencies between the use of methods and the creation of artifacts.
- A low number of different methods is sufficient to adequately cover the areas of feasibility, desirability, and viability, and to create sufficient artifacts.
- The use of methods and artifacts favors efficient and fast product development.

C. Research Method and Data Used

To approach the research topic, exploratory research was applied due to the unknown nature of the findings. The concept of exploratory research originally came from the social sciences, but is now also found in other areas, such as market research [41, p. 37]. It is used when there is no or little scientific knowledge about certain research subjects, but there is nevertheless a presumption of the existence of interesting elements [42, p. 6]. The most important characteristics and prerequisites for carrying out explorative research are flexibility in the search for data as well as openness and creativity during the research process [41, p. 37]. Frequently used methods are interviews, expert discussions or a literature search [41, p. 22, 42, p. 37].

A combination of qualitative and quantitative methods is used for the little-explored field of agile approach in the development of physical products. The starting point is a comprehensive literature research in order to create a common and uniform understanding of important basic concepts. The focus is also on agile procedures in the innovation processes of physical products.

The quantitative part is the data taken during the course Think.Make.Start of the Technical University Munich and the UnternehmerTUM, which took place at the UnternehmerTUM MakerSpace in Garching from 05.10 to 18.10.2016. During the course, the student teams were accompanied daily and their procedures documented. On the one hand, this systematically documented all activities such as applying methods, creating artifacts or the use of prototypes. On the other hand, the reasons and motivation for carrying out certain activities were determined by open and spontaneous individual and group interviews.

By combining the qualitative and quantitative results, a symbiosis takes place to establish a first generic concept. This corresponds to the qualitative part of the "partially known phenomenon" from Figure 5. The inductive creation of an agile framework and a logic for the structure of the components creates a qualitative basis on which ultimately a quantitative comparison of the collected data from TMS can take place. This corresponds to the deductive application and verification of the previously developed ideas.

The following diagram summarizes the implementation of the research methods in this work.

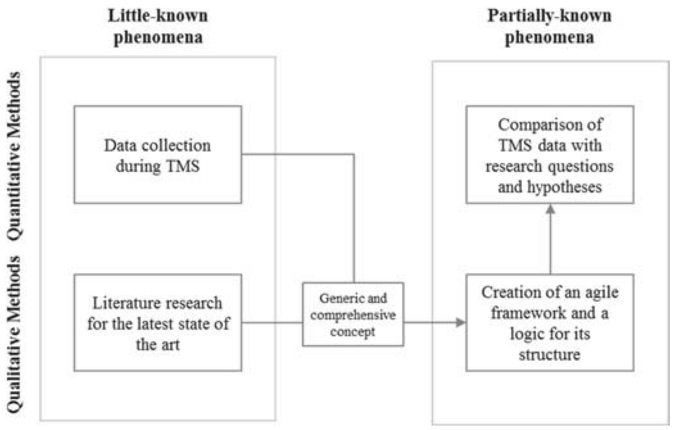

Fig. 5. Explorative research of this study according to [43, p. 6].

D. Data Collection

In order to present the results in a structured manner, the derived logic for the structuring of the components is transferred into a new schema (see Figure 6).

Methods and artifacts are now the focus, they take the upper and lower half of the schema. The characterization of the purpose is based on the three defined criteria. The "Why" corresponds to the step in the generic innovation process. Here, a further subdivision is partly made, in order to present the results in a more structured way. For example, the identification consists of the sub-categories "Problem definition and re-definition" as well as "Recognizing needs and synthesis". The "why" is shown in the upper part, the "what" in the lower part of the picture. The origin of the artifacts and methods, in short the corresponding procedure model, is evident by a different coloring. The associated legend is located below the image. Since the "when" refers to the corresponding iteration, it is not shown in the data base.

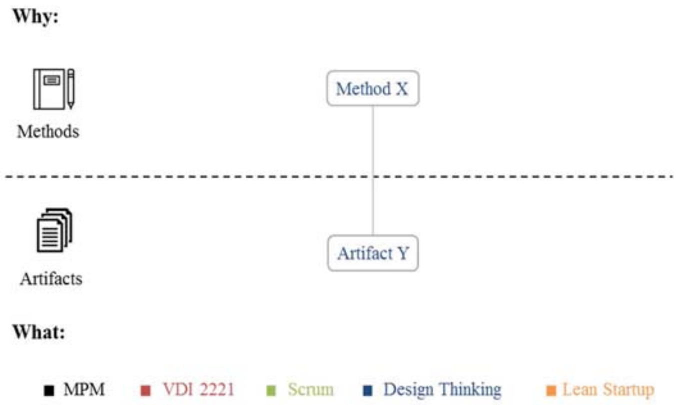

Fig. 6. Logic to fill the theoretic agile framework.

For collecting the data, interviews, observations, and questionnaires, as well as the documentations and presentations of the teams were used. Thereby, the created prototypes, their purpose, and the gained findings, were recorded. Further, the sequence and the links between the prototypes were documented. Therefore, each day the documents provided by the teams were analyzed. Data for describing the prototypes, the hypotheses tested with them as well as the gained findings were documented. In discussions with the team members, missing information and additional prototypes were queried. It proved difficult to gather data on software prototypes, as the programming was done mainly “quick-and-dirty” and thereby the single steps were not documented in a transparent version. Though, the focus of the data collection was on gathering all information possible about physical prototypes.

V. Results

A. Classification of Artifacts

Artifacts and methods can be categorized by their orientation to desirability, feasibility, and viability. Prototypes are divided into methods (manual, laser cut, rapid prototyping) and artifacts (function, geometry, concept). It is important that these orientations can also be assigned to the above-mentioned orientations. This is not the case for containers, since their purpose is to keep the project situation together. Figure 7 gives an overview of the clusters.

It turns out that many artifacts and methods can have different orientations. The affiliation is then dependent on the respective use case, for example the search can theoretically cover all three sub-areas. There is thus a further classification based on the practical use of TMS.

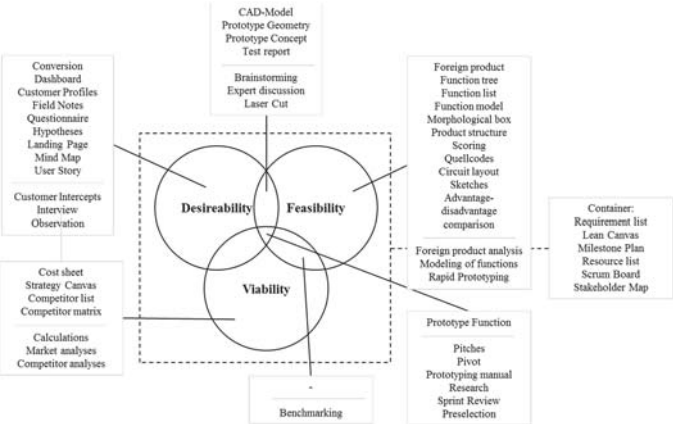

Fig. 7. Clustering of artifacts and methods into Desirability, Feasiblity, Viability and Container based on use cases during TMS.

B. Temporal Assignment

The temporal assignment consists of an allocation of the artifacts and methods for the respective iteration. Artifacts are identified when created, used, or modified. Methods in case of application.

With respect to the temporal distribution, with the exception of the last iteration, at least 10 different artifacts and 5 different methods were constantly used, created or modified. Many artifacts and methods are repeatedly in use over several iterations. These are therefore very important for the teams. The number of artifacts is still higher than that of the methods. This is explained by the fact that one method can result in multiple artifacts or artifacts can be created without using a method. The use of artifacts has a clear maximum in iterations 3, 4 and 5. The need for support in the form of codified and visual information (e.g. sketches or mind maps) is therefore initially higher than towards the end of the project when the product itself has already assumed concrete form. In contrast, the method use up to iteration 6 is rather constant. This underpins the hypothesis that a low number of different methods is sufficient to adequately cover the areas of feasibility, desirability, and viability, and to create sufficient artifacts.

C. Frequency of artifacts and methods in the stages of the generic innovation process

The frequency of the artifacts and methods occurring in each case is examined for further analysis. The results are shown in Figure 8. Each iteration is divided into two halves, artifacts are listed on the left and methods on the right.

Overall, the chart shows very clearly that the teams were actually agile during TMS. Partially, several stages of the generic innovation process are traversed in an iteration, for example, in iteration 3 it is even all six. Furthermore, there are also jumps within the process stages. For example, in the iteration 6, the identification of product ideas is repeated, although this activity is actually one of the first in the development of innovation.

The frequency of occurrence of artifacts and methods can be further divided into two larger blocks. The first block refers to the steps *identify*, *generate ideas* and *select the idea* and extends to iteration 4. Characteristic is that the innovation idea is focused here. The second block starts in iteration 4 and covers the areas of requirements / target / solving problem, solution alternatives and concept development as well as elaboration. Here, the concentration on the concrete product is very pronounced. Particularly interesting is the fact that the sub-section

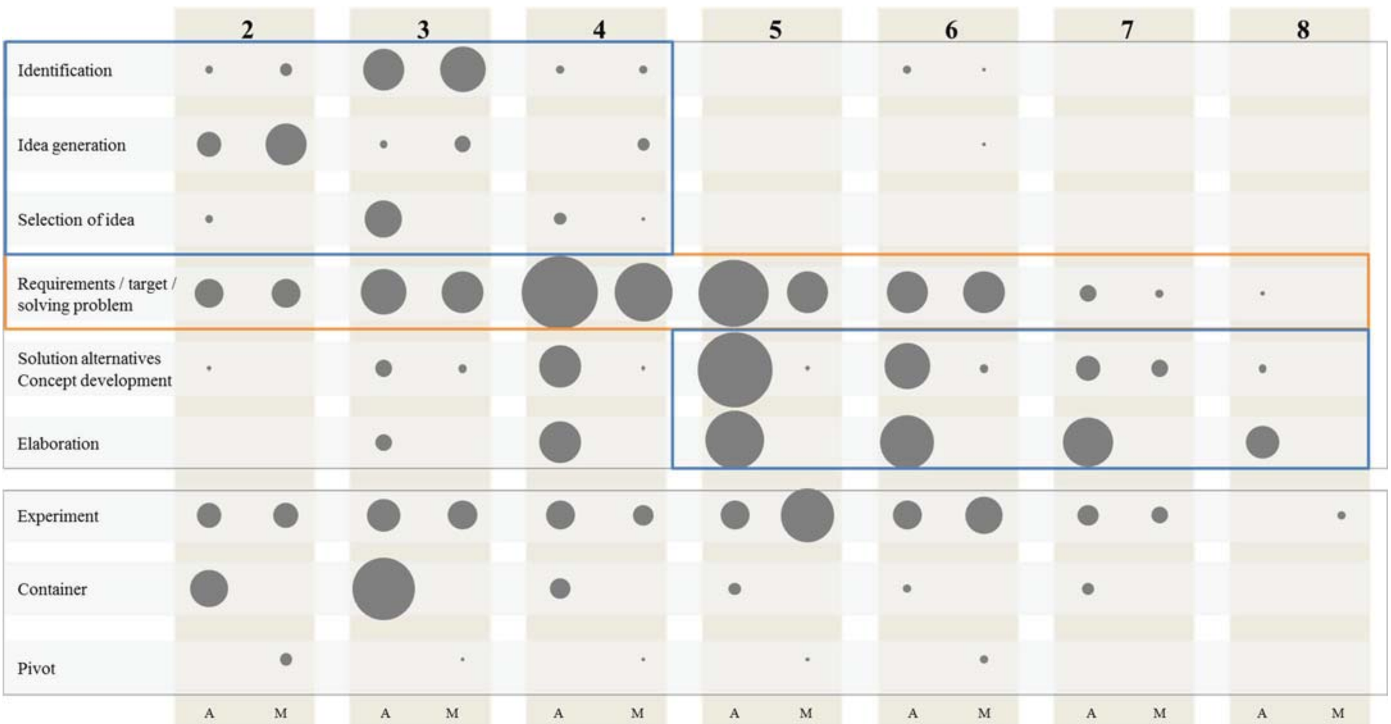

Fig. 8. Frequency of artifacts and methods within generic innovation process.

"requirements / target / clarify problem" can be referred to as a kind of connection or transmission element. The occurrence of artifacts and methods is already apparent from iteration 2 up to the end of the project. As mentioned above, most different artifacts and methods are used in this stage. It is therefore of great importance as a transformation element between the innovation idea and the concrete product.

Finally, it is stated that containers are created in early iterations. This is in the sense of its purpose to create a possibility of overview and cohesion. Furthermore, the production and use of prototypes as an experimental object is constant over all iterations. This is also in the sense of their actual goal of getting regular feedback.

In terms of frequency, the most important artifacts are source codes, questionnaires, hypotheses, sketches, and CAD models. The most common containers are the Lean Canvas and the Scrumboard. The rest of the artifacts are distributed over the iterations without any major conspicuousness. This is due to the fact that they were produced and used on the basis of specific project- or team-dependent characteristics and preferences. For example, a team often worked with a foreign product. The most popular methods are the interview, brainstorming and research. In addition, at least one pivot was carried out in the first 5 iterations. With regard to the experiments, manual prototyping and the use of the laser cutter were used to create conceptual and functional prototypes.

D. Analysis of Interdependecies – Example Bikorsa

In the following the example of team Bikorsa is outlined in detail (see Figure 9). The basic vision of Bikorsa described the development of an innovative new bicycle bag. For this purpose, a brainstorming for identifying customer problems was carried out in iteration 2, which was supplemented by a competition analysis for existing solutions. Furthermore, first interviews took place. The first own concept idea was visualized by sketches and converted into a concept prototype of cardboard by manual prototyping. During the day, the team also created a Lean Canvas to aggregate the findings so far.

An analysis of the first surveys led to the creation of a questionnaire and hypotheses in iteration 3, which were used in interviews. In order to structure the project work better, a Scrum Board was introduced. During a search for possible materials, a new competition analysis took place in parallel. This resulted in a competition list in combination with the results of Iteration 2. For the sake of clarity, mind maps and sketches were created, as well as a genuine bike (external product) for geometrical dimensions was included.

Based on this information, Bikorsa produced a new prototype with a new design. In Iteration 4, the conceptual prototype was used in the interviews. On the basis of the results, the questionnaire and the hypotheses were updated again.

Within the framework of a brainstorming, ideas were gathered as the team had not yet determined the desired functionalities. These were collected in a Mind Map. Due to the abundance of possible functions, the team felt insecure and conducted an expert discussion, which ultimately resulted in a pivot. In concrete terms, this meant that the team was limited to certain functionalities. At the same time, the first consideration was given to the technical feasibility and the necessary components, and a first CAD model for the fixture was made.

In addition, a list of requirements as well as a resource list was developed to collect historical findings. Due to the pivot, Bikorsa developed the product idea again in iteration 5 and restricted the target group. In this context a user story was created. In addition, the existing competition analysis was expanded and evaluated with regard to the new target group. This resulted in a competition matrix.

The product functions now required were collected in a function list and a morphological box. In addition, a part of the team researched solutions to existing brackets to refine the existing CAD model on this basis. The first product increment in the form of a functional fixture could now be presented by Laser Cutting. In addition, programming for the brake light was performed (second increment).

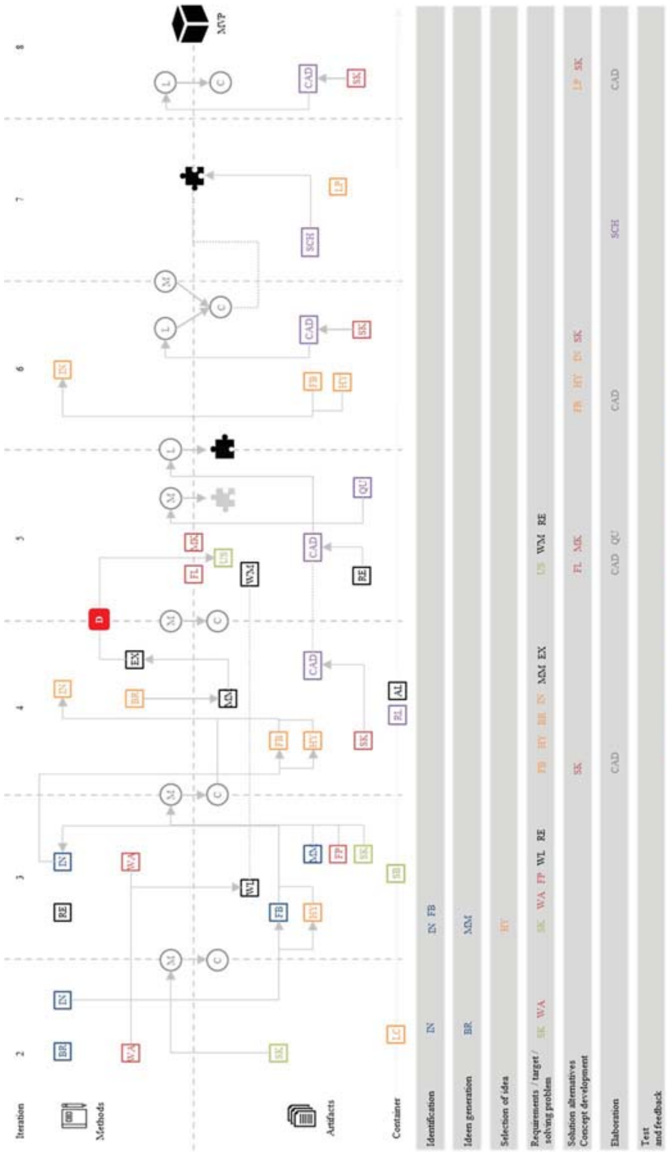

Fig. 9 Schematisc Illustration of the agile approach of team Bikorsa.

In Iteration 6, a new questionnaire and new hypotheses were designed to test the previously defined reorientation in interviews. As a result, the focus was placed on city bikers who attach importance to safety, user friendliness and design. For the evaluation of the design two different prototypes were built and reactions were tested. Iteration 7 was largely in the sense of technical refinement. Due to problems with the brake light a circuit diagram was created. In addition, Bikorsa designed a landing page. In the end, the previously developed bracket, the brake light and the actual cover could be combined into a functional overall concept. Iteration 8 served to finalize the docking station and the MVP created, while parallel preparations for the final presentation were running.

VI. Summary

This work analyzed the influence of artifacts and methods in agile development projects of physical product. In this area, there is little scientific knowledge, which is why the research methodology of exploratory research has been applied.

In order to make the research area more transparent, an examination of the current state of research was carried out. On the basis of extensive literature researches, important definitions were first made before different forms of expression were taken into consideration in product development. By examining specific approach models that are relevant to Think.Make.Start, a generic innovation process was derived. Due to the thematic importance, an analysis of the importance of artifacts in the product development took place.

Based on the results of the preceding sections, a first step towards an agile framework was developed, which is used in development projects of physical products. A further literature research led to the development of a logic for structuring the components of the framework. This is of great importance since this allows a systematic and uniform mapping of the agile development processes. The logic was then filled on the basis of the theory in order to create a data basis for analysis.

These concepts were then essential for a targeted analysis of the data from TMS. First of all, a comparison of the used artifacts and methods with the data collection from the theory was possible. The resulting logic allowed a systematic investigation of the temporal use, the affiliation to different stages of the generic innovation process as well as procedural models. Furthermore, a grouping into different clusters (desireability, feasibility and viability) was carried out. Finally, an analysis of the interdependencies between methods and artifacts took place. For this purpose, the innovation process of each individual team was presented clearly and structured in a uniform schema in order to identify any similarities or differences. This was supplemented by a textual description of the respective procedures. Finally, research questions or hypotheses were formulated, which are now being reviewed.

Hypothesis 1, 2 and 3 are confirmed. Hypothesis 4 is only partly confirmed. Artifacts and methods are necessary and quite useful. However, no general statement regarding their role can be made yet. Further research efforts are needed here.

VII. Discussion and Outlook

In order to systematically map the framework of Think.Make.Start, the agile framework was created. This was a necessary prerequisite for analyzing the components of agile development projects in a planned and clear manner. In its entirety, the TMS team's various approaches were outstanding. For applications in other agile projects it should be verified in more detail and adapted as required.

The framework was logically structured using its components. For this purpose, a literature research took place to identify existing approaches. Afterwards a symbiosis and the derivation of one's own logic took place. It should be noted that changes to the generic innovation process or the agile framework must also be adapted to their structure. Nevertheless, the logic provides a very good starting point for systematically mapping the different elements of an agile development.

In order to ensure a targeted analysis of the TMS data, a data base was necessary to control corresponding assignments and comparisons. Such a data base is not available in research and is based on artifacts and methods from the investigated approach models. It can be extended and supplemented at any time by further analysis procedures. However, the data developed here provided an extensive basis for a systematic investigation of the development projects.

Data collection was very broad, including quantitative and qualitative data, e.g. in the course of spontaneous surveys of the team members. This combination of data was finally evaluated in combination with the knowledge and concepts developed in the course of the work.

In particular, the logic for structuring the components of the framework proved to be extremely suitable for obtaining insights and for making first derivations. The analysis of the interdependencies emerged as a more complicated one as a result of the very individual approach of the teams. However, a systematic mapping of each individual development project over the entire iterations provided a suitable foundation for carrying out initial analyzes. Further research efforts are useful here to obtain more detailed insights.

Acknowledgment

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.

This project has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No 720270 (Human Brain Project SGA1)

References

[1] P. Link, “Agile Methoden im Produkt-Lifecycle-Prozess – Mit agilen Methoden die Komplexität im Innovationsprozess handhaben,” in *Komplexitätsmanagement in Unternehmen*, K.-P. Schoeneberg, Ed., Wiesbaden: Springer Fachmedien Wiesbaden, 2014, pp. 65–92.

[2] P. Link and M. Lewirck, “Agile methods in a new area of innovation management,” *Science-to-Business Marketing Conference*, 2014.

[3] A. Böhmer *et al.,* “Think.Make.Start. - An Agile Framework,” in *14th International Design Conference Dubrovnik, 16-19 May 2006*, D. Marjanovic, M. Storga, N. Pavkovic, N. Bojcetic, and S. Skec, Eds., Glasgow, 2016, pp. 917–926.

[4] M. R. Gürtler and U. Lindemann, “Innovationsmanagement,” in *Handbuch Produktentwicklung*, U. Lindemann, Ed., München: Hanser, 2016, pp. 483–512.

[5] V. Krishnan and K. T. Ulrich, “Product Development Decisions: A Review of the Literature,” *Management Science*, vol. 47, no. 1, pp. 1–21, 2001.

[6] B. Oestereich, “Agiles Projektmanagement,” *HMD*, vol. 260, pp. 18–26, 2008.

[7] R. G. Cooper, “Agile-Stage-Gate Hybrids,” *Research Technology Management*, vol. 59, no. 1, 2016.

[8] N. Ovesen and A. F. Sommer, “Scrum in the Traditional Development Organization: Adapting to the Legacy,” in *Modelling and Management of Engineering Processes*, M. Schabacker, K. Gericke, N. Szélig, and S. Vajna, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 87–99.

[9] M. Kirchhof and B. Aghajani, “Agil in die Sackgasse,” *Manage it*, vol. 8, no. 9, pp. 1–6, 2010.

[10] T. S. Schmidt and K. Paetzold, “Agilität als Alternative zu traditionellen Standards in der Entwicklung physischer Produkte: Chancen und Herausforderungen,” in *Design for X - Beiträge zum 27. DfX-Symposium Oktober 2016*, D. Krause, K. Paetzold, and S. Wartzack, Eds., Hamburg: TuTech Verlag TuTech Innovation GmbH, 2016, pp. 255–267.

[11] Duden, *Artefakt, das.* [Online] Available: http://www.duden.de/rechtschreibung/Artefakt. Accessed on: Dec. 19 2016.

[12] Object Management Group, “OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2: OMG Available Specification without Change Bars,” 2007. [Online] Available: http://doc.omg.org/formal/2007-11-02.pdf. Accessed on: Dec. 19 2016.

[13] B. Rumpe, *Agile Modellierung mit UML: Codegenerierung, Testfälle, Refactoring,* 2nd ed. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2012.

[14] A. Fay, M. Schleipen, and M. Mühlhause, “Wie kann man den Engineering-Prozess systematisch verbessern?,” *atp edition*, vol. 51, no. 01-02, pp. 80–85, 2009.

[15] T. Brökel, *Wissens- und Innovationsgeographie in der Wirtschaftsförderung: Grundlagen für die Praxis*. Wiesbaden: Springer Gabler, 2016.

[16] T. Gülke, “Erweiterung des Anforderungsmanagement-Fokus: Von Produkten zu Prozessen,” Dissertation, Fakultät für Mathematik, Informatik und Naturwissenschaften, RWTH Aachen, Aachen, 2014.

[17] T. P. Klein, “Agiles Engineering im Maschinen- und Anlagenbau,” Dissertation, Institut für Werkzeugmaschinen und Betriebswissenschaften (iwb), Technische Universität München, München, 2016.

[18] H. Balzert, *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements-Engineering,* 3rd ed. Heidelberg: Spektrum Akademischer Verlag, 2009.

[19] A. Hahn, S. Häusler, and S. große Austing, *Quantitatives Entwicklungsmanagement: Modellbasierte Analyse von Produktentwicklungsprozessen*. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2013.

[20] T. Fischer, H. Biskup, and G. Müller-Luschnat, “Begriffliche Grundlagen für Vorgehensmodelle,” in *Teubner-Reihe Wirtschaftsinformatik, Vorgehensmodelle für die betriebliche Anwendungsentwicklung*, R. Kneuper, G. Müller-Luschnat, and A. Oberweis, Eds., Leipzip: B. G. Teubner Verlagsgesellschaft, 1998, pp. 13–31.

[21] U. Hammerschall, “Flexible Methodenintegration in anpassbare Vorgehensmodelle,” Dissertation, Institut für Informatik der Technischen Universität München, Technische Universität München, München, 2008.

[22] K. Hoffmann, “Projektmanagement heute,” *HMD*, vol. 45, no. 2, pp. 5–16, 2008.

[23] T. P. Klein and G. Reinhart, “Towards Agile Engineering of Mechatronic Systems in Machinery and Plant Construction,” *Procedia CIRP*, vol. 52, pp. 68–73, 2016.

[24] W. W. Royce, “Managing the development of large software systems: concepts and techniques,” *IEEE Computer Society Press*, pp. 328–338, 1987.

[25] B. W. Boehm, “A spiral model of software development and enhancement,” *IEEE Computer Society Press*, vol. 21, no. 5, pp. 61–72, 1988.

[26] F. Bodendorf, *Computer in der fachlichen und universitären Ausbildung*. München: Oldenbourg, 1990.

[27] M. A. J. Gnatz, “Vom Vorgehensmodell zum Projektplan,” Dissertation, Fakultät für Informatik der Technischen Universität München, Technische Universität München, München, 2005.

[28] J. Goll and D. Hommel, *Mit Scrum zum gewünschten System*. Wiesbaden: Springer Vieweg, 2015.

[29] K. Gurusamy, N. Srinivasaraghavan, and S. Adikari, “An Integrated Framework for Design Thinking and Agile Methods for Digital Transformation,” in *Lecture Notes in Computer Science, Design, User Experience, and Usability: Design Thinking and Methods*, A. Marcus, Ed.: Springer International Publishing Switzerland, 2016, pp. 34–42.

[30] J. Coldewey, “Agile Entwicklung Web-basierter Systeme: Einführung und Überblick,” *Wirtschaftsinformatik*, vol. 44, no. 3, pp. 237–248, 2002.

[31] A. Cockburn, “Agile Software Development Joins the "Would-Be" Crowd,” *Cutter IT Journal*, vol. 15, no. 1, pp. 6–12, 2002.

[32] L. Cao, K. Mohan, P. Xu, and B. Ramesh, “A framework for adapting agile development methodologies,” *European Journal of Information Systems*, vol. 18, no. 332-343, 2009.

[33] B. W. Boehm and R. Turner, *Balancing Agility and Discipline: A Guide for the Perplexed*. Boston: Addison-Wesley, 2004.

[34] M. Kuhrmann, D. M. Fernandez, and M. Grober, “Towards Artifact Models as Process Interfaces in Distributed Software Projects,” in *2013 IEEE 8th International Conference on Global Software Engineering (ICGSE)*, Aug. 2013, pp. 11–20.

[35] H. Femmer, M. Kuhrmann, J. Stimmer, and J. Junge, “Experiences from the Design of an Artifact Model for Distributed Agile Project Management,” in *2014 IEEE 9th International Conference on Global Software Engineering (ICGSE)*, Aug. 2014, pp. 1–5.

[36] J. M. Bass, “Artefacts and agile method tailoring in large-scale offshore software development programmes,” *Information and Software Technology*, vol. 75, pp. 1–41, 2016.

[37] O. Liskin, “How Artifacts Support and Impede Requirements Communication,” in *Lecture Notes in Computer Science, Requirements Engineering: Foundation for Software Quality*, S. A. Fricker and K. Schneider, Eds., Cham: Springer International Publishing, 2015, pp. 132–147.

[38] D. Méndez Fernández, S. Wagner, K. Lochmann, A. Baumann, and H. de Carne, “Field study on requirements engineering: Investigation of artefacts, project parameters, and execution strategies,” *Information and Software Technology*, vol. 54, no. 2, pp. 162–178, 2012.

[39] J. Menold, T. W. Simpson, and K. W. Jablokow, “The Prototype for X (PFX) Framework: Assessing the Impact of PFX on Desirability, Feasibility, and Viability of End Designs,” in *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Sunday Aug. 2016, V007T06A040.

[40] U. Lindemann, *Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden,* 3rd ed. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2009.

[41] H. Böhler, *Marktforschung,* 3rd ed. Stuttgart: Kohlhammer, 2004.

[42] R. A. Stebbins, *Exploratory research in the social sciences*. Thousand Oaks, California: SAGE Publications, 2001.

[43] W. B. Shaffir and R. A. Stebbins, Eds., *Experiencing fieldwork: An inside view of qualitative research*. Newbury Park: SAGE Publications, 1991.