AGILE DEVELOPMENT OF PHYSICAL PRODUCTS

An Empirical Study about Motivations, Potentials and Applicability

Tobias Sebastian Schmidt, Stefan Weiss, Kristin Paetzold

2017

iPE Institut für Technische Produktentwicklung
AGENSIS MANAGEMENT CONSULTANTS
Universität der Bundeswehr München

Cite as:
Schmidt, T. S., Weiss, S., and Paetzold, K. (2018), *Agile Development of Physical Products: An Empirical Study about Motivations, Potentials and Applicability*, ISBN: 978-3-943207-28-6, available at: www.unibw.de/produktentwicklung-en, Report, University of the German Federal Armed Forces, Munich, Germany.

We love feedback:
Please do not hesitate to give us feedback by contacting *agile@unibw.de*. Feedback helps us to improve research on agile development of physical products. We constantly work on developing it further.

ISBN: 978-3-943207-28-6

Publisher:
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39
85577 Neubiberg, Germany

Front cover photo credits: istockphoto.com: PeopleImages, archerix

Acknowledgments

As the author team, we would like to express our sincerest gratitude to all those who supported us in preparing, conducting, analyzing and documenting this study. A special gratitude we give to the survey participants. The survey contained 40 questions and most participants spent more than 30 minutes to answer all questions. This work is extraordinarily extensive for a research survey. We very much appreciate the trust and confidence the participants put on us. Giving such deep and detailed insights into industrial practice is highly valuable for research. Without their contribution, this work would not have been possible in no case. Furthermore, we would also like to acknowledge with much appreciation the support of Verein Deutscher Ingenieure e.V. (Association of German Engineers) and their staff who facilitated a large scale mailing list with more than 16,000 potential survey participants.

Executive Summary

In order to cope with volatile, uncertain, complex and ambiguous (VUCA) conditions in product development, more and more companies want to increase their agility. While agile development has become a standard in software development especially in the last decade, the development of hardware still follows traditional procedures such as the V-model. As less reliable, quantitative facts exist for agile development of physical products in particular (hardware), the study at hand aims at shedding light on reasons why to implement agile hardware development, actual improvements of agile hardware development, and applicability of the concepts that were actually designed for agile software development. In order to interpret the survey answers correctly, the study also analyzes what companies actually mean when talking about agility in product development.

Survey design and demographics
Supported by Verein Deutscher Ingenieure e.V. (Association of German Engineers), the author team conducted a broad online survey containing 40 questions. To reduce biases due to cultural differences or translations, the survey was set up in German language and sent to companies from Germany, Austria and Switzerland. In total, 228 persons from various job positions and focuses of work participated. Among the participants are especially large companies (ca. 40% have more than 5,000 employees and more than 500 m€ annual sales) particularly from mechanical and plant engineering (32%), as well as vehicle and traffic engineering (19%).

Results about understanding
When talking about agility in product development, practitioners mean especially a versatile (26%), lively (22%) and lean (16%) project organization. While most participants agree that agile development is about self-organization (4.4/5), team's decision authority about technical solutions (4.1/5) and willingness to honestly disclose the current project progress towards the customer (4/5), most disagree that agile development means to let the team grant themselves administrative permits like vacation requests (2.7/5), to work without a specification sheet signed by all parties (2.6/5) and to apply Scrum (2.4/5). Scrum (32%), Kanban (20%) and Design Thinking (16%) as well as associated practices are starters for agile hardware development. The more experienced a company becomes, the more practices it uses that support operative project work (designing instead of organizing, coordinating and steering).

Results about motivations
The reasons why companies start to implement agile hardware development are manifold. More than 75% of the participated companies said that they want to improve communication, reduce reaction time to changes, increase project effectiveness, increase flexibility and shorten time-to-market. Classical key performance indicators like improved adherence to schedules, shortened time-to-market and increased productivity are among the Top 7. In contrast, much less than 50% of the participated companies aim at improving internal learning and knowledge creation, reducing development costs and improving customer understanding.

Results about potentials
Experienced companies (midcourse, advanced or completed in implementation) report that the

real value of agile hardware development is improved communication (3.2/4), increased transparency (3.2/4), reduced reaction time to expected and unexpected changes (3.2/4) and increased flexibility (3.1/4). It reveals that especially soft factors, that are hard to monitor quantitatively, actually improve more than classical key performance indicators. Among 23 tested partial values of agile hardware development, reduced development costs, advanced product quality, improved development processes, shortened product development and improved adherence to schedules are within the last 11 ranks. A gap between desired and real effects of agile hardware development is present in the data. Especially improved adherence to schedules, shortened time-to-market and reduced costs is hyped (overestimated). In contrast, increased transparency, improved internal learning and knowledge creation, and increased project-related commitment turn out to be most underestimated.

Results about applicability
The more experienced the participants are, the higher they rate the transferability of the Manifesto from agile software to agile hardware development. In total, about 60% of all participants say that the Manifesto is quite well or well transferable. However, challenges to become agile in hardware development are especially in establishing an agile working attitude (3.0/4) and embedding agile teams in classical organized companies (2.8/4). Surprisingly, almost all tested challenges are overestimated; during implementation, they turn out to be less challenging than expected. Furthermore, more than two thirds of participated companies start with agile software development before adopting it for hardware, too. The chance to reach an advanced or completed implementation progress within the first two years is less than 25%.

Conclusion
Agile hardware development is an appropriate approach to tackle dynamic conditions in industrial practice. However, many assumptions and myths such as *"Agile development makes the project cheaper and quicker"* exist in the field of agile hardware development. Some of them could be true for a few cases, but the study proves that not all of them can be generalized. Agile hardware development is beneficial especially for VUCA environments. However, companies should think twice, if they want to become agile because it is trendy, or if their context conditions really require more transparency, flexibility and shorter reaction time to respond to foreseeable and unforeseeable changes.

About the Author Team

M.Sc. Tobias Sebastian Schmidt (ITPE) has studied mechanical engineering and engineering management in Germany, Finland and Sweden, and graduated with a Master of Science at the Technical University of Hamburg. Fascinated by innovation and new product development, he started a Ph.D. in agile hardware development at the University of the German Federal Armed Forces Munich in 2015. At the Institute of Technical Product Development (ITPE), he investigates how – and to which degree – agile development approaches can be transferred from software development to the development of physical products such as mechatronic systems. In his research, he focuses especially on prototyping and the so-called constraints of physicality. In this context, he has published several research papers and has given talks on international conferences about the challenges and potential solutions.

Dr.-Ing. Stefan Weiss (AGENSIS Management Consultants) is Partner and cofounder of AGENSIS management consultants. Dr. Weiss has more than 20 years of professional experience in R&D Management and the design of holistic product development systems. He worked with numerous national and international clients and project teams from automotive, electronic, automation and consumer industry. In his work, he pioneered and applied the concepts of agile and lean development to R&D improvement programs for physical product development. In recent years, he is co-authoring several publications concerning lean and agile development. He has a profound industrial experience from different positions in R&D organizations of automotive and consumer industries. He graduated at Karlsruhe Institute of Technology (Germany) with a diploma in chemical engineering followed by a doctorate in material sciences.

Prof. Dr.-Ing. Kristin Paetzold (ITPE) is head of the Institute of Technical Product Development at the Faculty of Aerospace Engineering at the University of the German Federal Armed Forces. After completing her doctorate, Dr. Paetzold worked as a senior engineer at the Chair of Design Engineering at the University of Erlangen, focusing on research in the fields of mechatronics especially in supporting and optimization of development processes. There she started to set up a working group dealing with challenges in describing, analyzing and optimizing development processes based on data and information flows. With becoming a full professor at the University of the German Armed Forces in 2009, the research focus has expanded to include systems engineering. She is involved in the VDI department of product development and mechatronics and is also Co-Chair of the SIG "Human Behavior in Design" of the Design Society.

Foreword by Reiner Köttgen

Expert Agile Transition, TRUMPF GmbH & Co. KG

Agile resounds throughout the country. Agile projects, Agile methodologies, Agile organizations, Agile enterprises, Agile work schedules, the list could be continued arbitrarily. Challenges that the software industry has approached for 20 years are becoming part of our daily personal as well as corporate life, be it Business-to-Business or Business-to-Consumer, producing or providing services: more and more complex requirements change faster continuously. One of the most important domains of the German economy, the machine building engineering industry, recognizes increasingly the same phenomena for its markets.

When in 2014 TRUMPF GmbH & Co. KG, a market leading German laser technology and machine tool developer and manufacturer, started to develop machine tools using the Agile methodology Scrum, we were considered exotic. Not even 4 years later, the number of conferences and workshops on these topics, particularly designed for the hardware industry, rockets, the number of companies asking to visit our Research & Development department to learn about Agile in the manufacturing industry have increased rapidly.

To master these changes, the machine building industry is looking for new ways to develop the right product in an efficient manner, hence, defines new frameworks and develops matching methods, assumes a new stance and mindset, even talks about changing culture. TRUMPF already adopted Agile methodologies in more than half of its hardware development projects for machine tools. The goal is for any new project to be managed in an Agile manner. The following example demonstrates that this is a valid alternative to standard project management methods, at times the only way to master the complexity of the products to be developed: In 2014 TRUMPF made a far-reaching experience. The vision of an extensively new machine was very clear to the development team. Yet, the complexity of the challenge to translate it into a marketable machine was too big for the team to grasp: “With the development and project management methods at hand we find ourselves not capable of developing this machine!” Through external consulting the project was relaunched, Scrum was introduced as the project management method, the requirements were prioritized as well as iteratively and incrementally developed. At the Euroblech 2016 fair, the machine performed flawlessly throughout the whole exhibition, although it had been declared not developable once. We take the opportunity to build upon the expertise the software industry has gained throughout the past 20 years, to copy their models and procedures, and derive our own insights. In doing so, we also question existing organizational structures as well as processes, e.g. we redesign the project and product portfolio management processes accordingly.

This requires a considerable amount of courage. Many corporations are built upon decades - if not centuries - of their own tradition with an extraordinary proven record of success. Just these days, in early 2018, order books are full to an extent that exceeds production capacities. Critical questions to potential changes in corporate leadership and action do not surprise. “Why should we leave our successful path?” “There must be something we are doing well!” “Why taking the risk of a major transformation, let alone a cultural change?” On the other hand, changing behavior and operation of corporations will create pressure e.g. on suppliers: they will need

to be in a position to react to changing requirements on short cycles, ideally phase in to the rhythm of the development. Despite all applied courage, there are limitations. Although even the Business-to-Business market excepts continuous updates to software (for security flaws even daily), no TRUMPF customer will allow for service technicians to show up every three weeks, to disrupt the production in order to mount new functions to their machine tools.

Out of the operationalization of the Agile methodologies in day-to-day hardware projects new questions arise. In software, a few lines of code for a new app-feature can be developed in the morning, through fully automated testing be qualified in the afternoon and for lack of production be deployed to some millions of smartphones at night. However, the hardware of the machine building industry has its limits due to the machine's physics, the complexity of production and the duration of installation and launch to production – or as it is paraphrased in this study: The constraints of physicality.

Initially, Agile methods such as Scrum etc. were conceived in and framed for software development. TRUMPF's experience shows that these methods need further development and adaption, need a certain scale of freedom and flexibility to live up to the differences in each products, solutions and markets. In the context of the study at hand, leading corporations of the respective industry, that have gone in an Agile direction through extensive experiments and trials, and have even implemented it into practice, confirm exactly the above-mentioned need for methodological refinement.

Implementing Agile methods is everything but easy. Besides courage and endurance, it requires guidance and the openness to change among all stakeholders. By taking a neutral, scientific perspective, this study gives companies a great opportunity to understand what agility is really about before implementing it. I wish we would have had such an investigation at hand when we started. Quantitative facts from a neutral research standpoint give guidance and reduce the risk of being affected by the hype.

It is a safe assumption that the future will look different from the present ever faster. If we, the machine building industry, disengage from our gaze, misty-eyed by our current successes, we will notice that in the past 50 or 100 years the German machine building industry achieved its prosperity only through steady adaptions and improvements. By the dimension of challenges we are facing, these steady modifications will not be sufficient. A drastic change will be necessary. Meeting the accruing challenges, we will be strengthened and will emerge from this change even more successful. An Agile stance and mindset will enable us to identify the proper answers on all organizational levels and to continuously reiterate them depending on the current situation.

TRUMPF

Reiner Köttgen
Expert Agile Transition
TRUMPF GmbH & Co. KG

Contents

Part I.

Introduction

1. Goals and Motivation of the Study

In the organization and management of product development, the concept of agile product development has gained growing interests in recent years. On the one hand, the interests are fired by the increasing number of published success stories. They report that agile development can shorten the time-to-market and increase the chance that the developed product fits the real customer needs more accurately. Especially the software industry and start-ups provide many examples here.

On the other hand, traditional industries are challenged by future scenarios and expectations for new markets and products, such as the Internet of Things, Automated Driving, Industry 4.0, Smart Grid, Smart Home, and so on. Such evolving changes, for instance, cause increasing volatility, uncertainty, complexity and ambiguity (VUCA). In order to survive as a company under VUCA conditions, methods and processes from software development seem to be promising for hardware industries, too. Although skepticism exists whether agility can also realize its potential in the world of product development characterized by physical contradictions and constraints (Constraints of Physicality), the euphoria seem to outstrip the skepticism for many companies. The concept of agile development and the methods anchored in it are expected to offer significant potential for increasing competitiveness under ever-changing environments.

Nevertheless, high expectations can lead to over-ambitious goal settings and, if the desired effects do not set in immediately, the approach could be dropped (hype effect). Likewise, the full potential can only come into play, if the methods are used reasonably and appropriately, which requires learning curve effects and good guidance.

In light of these two outlined perspectives (constraints of physicality and hype effect), the need for research was derived to capture the current state of application of the concept of agile development in companies with physical product development. Therefore, the study aims at supporting and underpinning the decision-making process for the implementation of agile product development of physical products by providing statistical facts.

For that, the study examines the following aspects in detail:

Motivation: What do companies hope to achieve when they implement agile development of physical products?

Potentials: What can companies really achieve with agile development of physical products and is agility in the hype status of inflated expectations?

Applicability: Is it feasible to apply the agile approach from software industries to the development of physical products or are associated values, principles, methods and practices limited to agile software development?

The study focuses exclusively on the development of physical products that were defined as products that consist of mechanics, electronics or firmware at least to some extent. Thus, such products have a physical, tangible component and are not purely made of (virtual) software. In the context of this study, the term hardware is used interchangeably to physical product.

Without anticipating the results in detail, the data analysis shows partially surprising, partially alarming results. The potentials of the agile development concept are clearly present for physical product development, but the intensity and ranking are remarkable.

Finally, the author team once again likes to thank all participants, who spent their time in answering the extensive survey. Without their engagement these interesting results would not have been possible.

2. Survey Design and Execution

To capture the current state of agile hardware development, the author team designed a questionnaire that was published as an online survey. The survey design bases on experience from industrial projects and research, which the cooperation partners provided to the study team.

The study was conceived as a German-language study to prevent cultural or translation biases. All participants are from the German-speaking countries Germany, Austria and Switzerland (D/A/CH). The online survey was open from June to November 2017. In total, 228 practitioners participated in the study. Although the study was conceived as a German-language study and therefore did not capture "multilingual international experience" directly, the evaluations on companies and project experience (see following pages) show that an "internationality of development experience" is indeed present in the data.

Potential participants for the study were approached by industry and cooperation partners of the author team as well as by a call for participation by VDI (Verein Deutscher Ingenieure e.V.), the largest association of engineers in Germany, Austria and Switzerland. The VDI invitation was part of monthly newsletters in order to reach the widest possible target group.

The survey included 40 questions divided into 7 sections. Through the online process, the order of answer options were randomized for each participant in order to avoid desirability biases.

The questions were designed in different formatting types. Questions could be answered by yes / no, approval scales (1-5) or by free text. Which question format was used is also shown, where necessary, in the graphs or text of the result sections. The participants responded voluntarily and without direct address or assistance from the author team.

A special feature of the study was the concept of recording the "before – after" experience, which was transposed in the sequence of the question sections: The participants, who signaled at the beginning of the questionnaire that they are only at the beginning of an agile product development implementation were asked for their expectations, but not to their experiences. Thus, the online process skipped the section "experiences made" when the participant was a beginner. Consequently, a sample of participants could be generated which has clearly gained experience before and after implementation. As a side effect, each participant only had to spend the time that was really necessary.

The entire evaluation was done on anonymous data. The extracted raw data was digital through the online tool and was subjected to a consistency check. Responses that were incomplete or obviously not related to physical product development were removed from the evaluation. Selection criteria for this were:

- Product is a pure software one (stand-alone, PC, etc.)
- Industry affiliation is non-manufacturing industry (finance, construction, etc.)
- Lack of product development within the company (e.g. no development staff)

Part II.

Demographics

3. About the Respondents

To classify and interpret the answers, data on the participants and their company context are necessary. The following priorities have been set in the survey:

- Questions about the company and its situation
- Questions about the tasks and experience in the company
- Questions about the product structure
- Questions about the organization of product development

The first two question clusters (company & tasks and experience) are standard for any type of survey. However, the last two clusters (product structure & development organization) are unique in the context of agile development. To the author team, these topics appeared to be crucial in order to investigate the development of physical products and had to expect "constraints" or "special experiences" from these perspectives, which could make the difference to software development apparent.

Demographic aspects for all participants, regardless of which implementation progress they have, were evaluated in total.

3.1. Company-related Questions

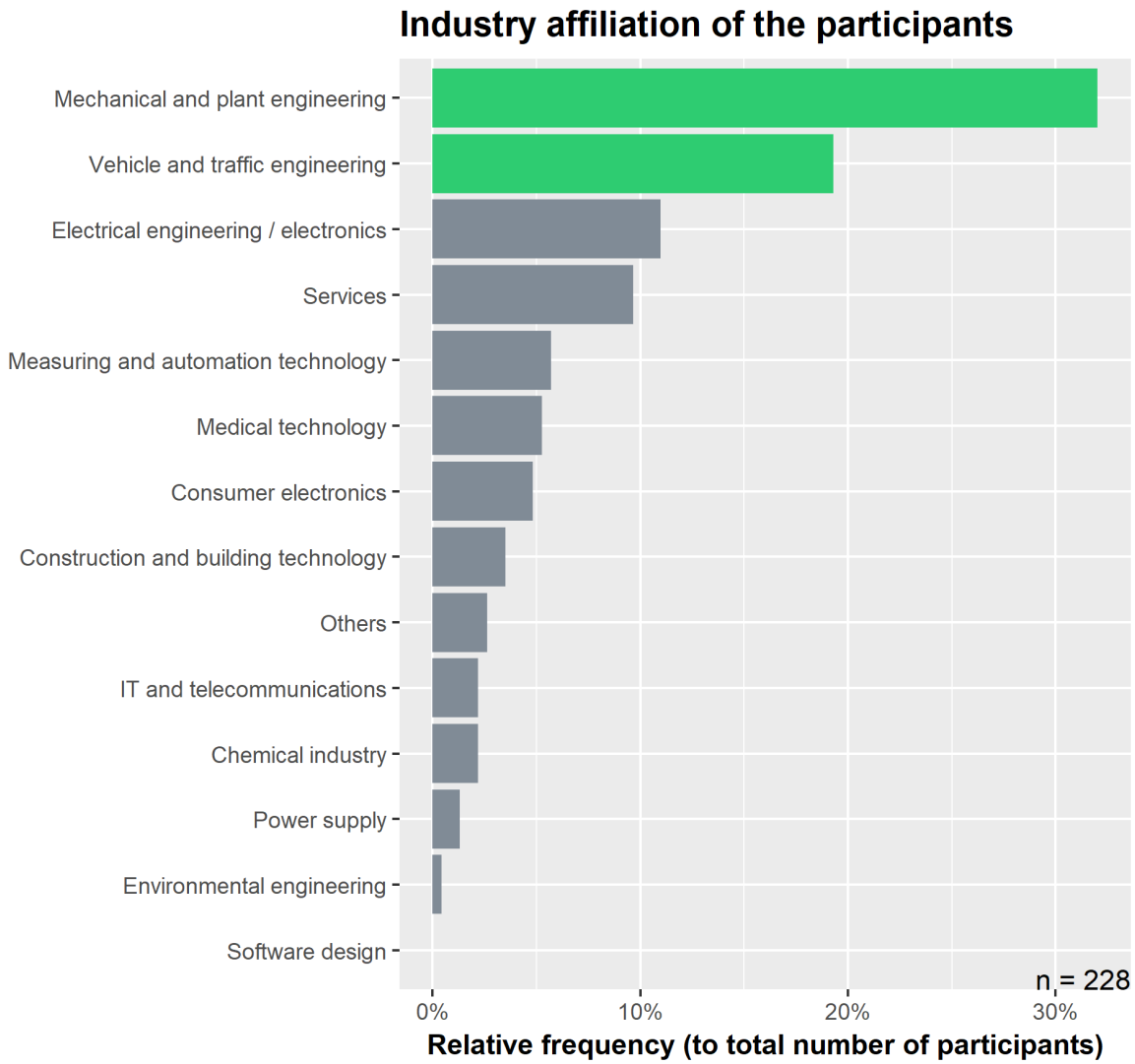


Figure 3.1.: Industry affiliation of participated companies.

Description
The categories of industry affiliations were specified. If participants could not assign themselves, they were allowed to give a textual description in their own words. These descriptions were later checked by the team and reassigned to a category, if possible.

Figure 3.1 shows the distribution of the participants across the industries. Significant proportions of participants come from mechanical engineering with 30%, followed by automotive industry with 18%. The noticeably large, unspecific proportion of services (13%) also has to be classified as product development-related. The category services contain some consultancies but also service providers, which develop products for others as a service.

Key learnings
- The cross-sectional span of the industrial sectors the study aimed at is generated.
- Vehicle industry is recognizable - mechanical engineering is dominant.
- Pure software companies are almost unrepresented, which was a goal of the study.

Interpretation

- The targeted study focus on physical product development is realized.
- Manufacturing industry has a clear interest in the topic.
- Agility, although originating from software industries, is perceived as potential approach for the optimization of product development and project management also for physical products.
- Large volume production and development industries are in place, but the experiences from mass production do not dominate.

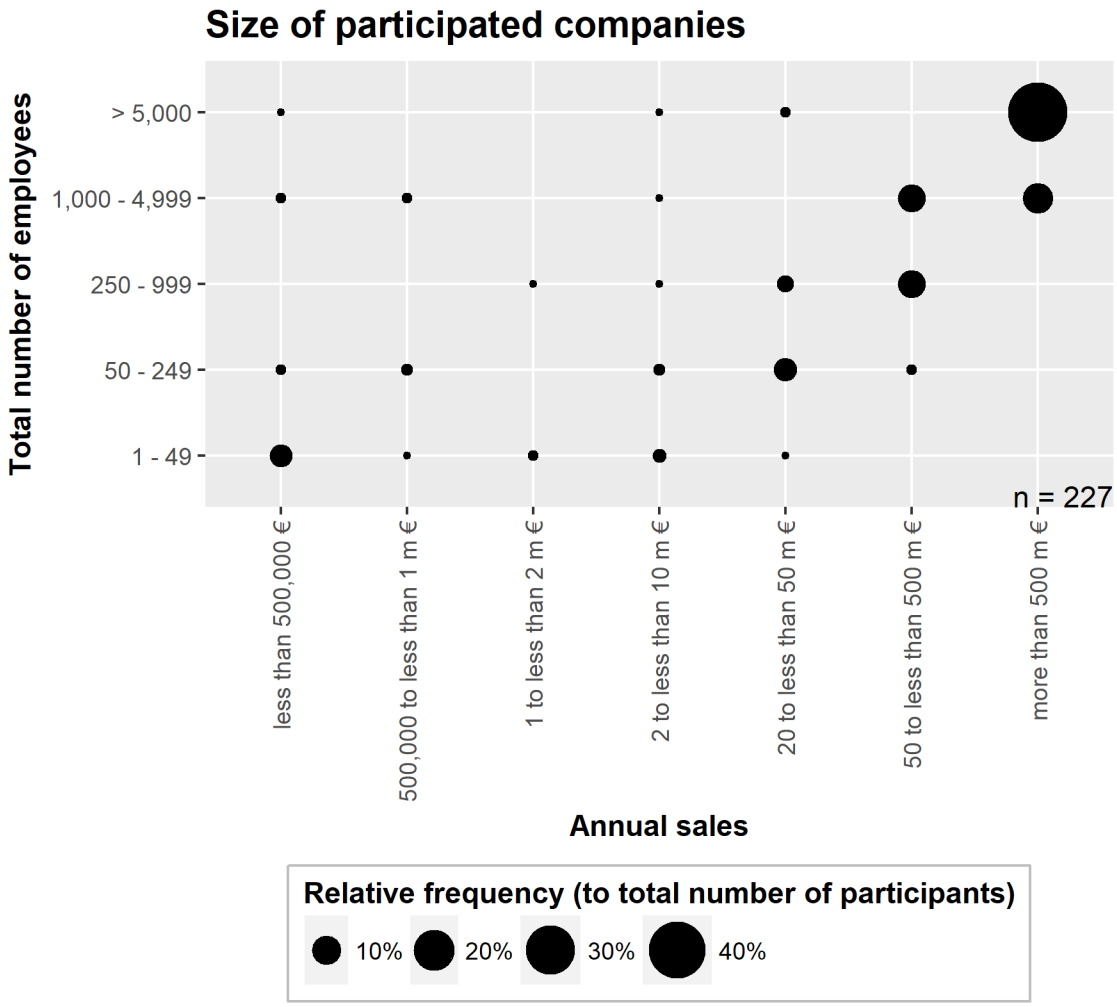


Figure 3.2.: Size of participated companies measured by annual sales and total number of employees.

Description
Figure 3.2 shows the distribution of company size according to turnover and number of employees. In the survey, the classes were given. The participants could not provide any other information. The breakdown of companies by size shows that there is a large span. In addition to large companies (share of approx. 50%), numerous SME's are also represented.

Key learnings
- Large companies dominate with more than 40% (upper right quadrant).
- Typical SME's are represented with approx. 20% . Not only large companies find the agile development concept interesting and are actively involved in it.
- The upper left quadrant does not seem to be plausible. They might be outliers.

Interpretation
- The company size is not a limiting criterion for the application or the attractiveness of agile development.
- The fact that large companies are proportionally stronger represented corresponds to the expectations from previously published studies.
- The broad distribution is an indication that the study is a cross-section of the current corporate landscape - there is no centering on global leaders or publicly listed companies.

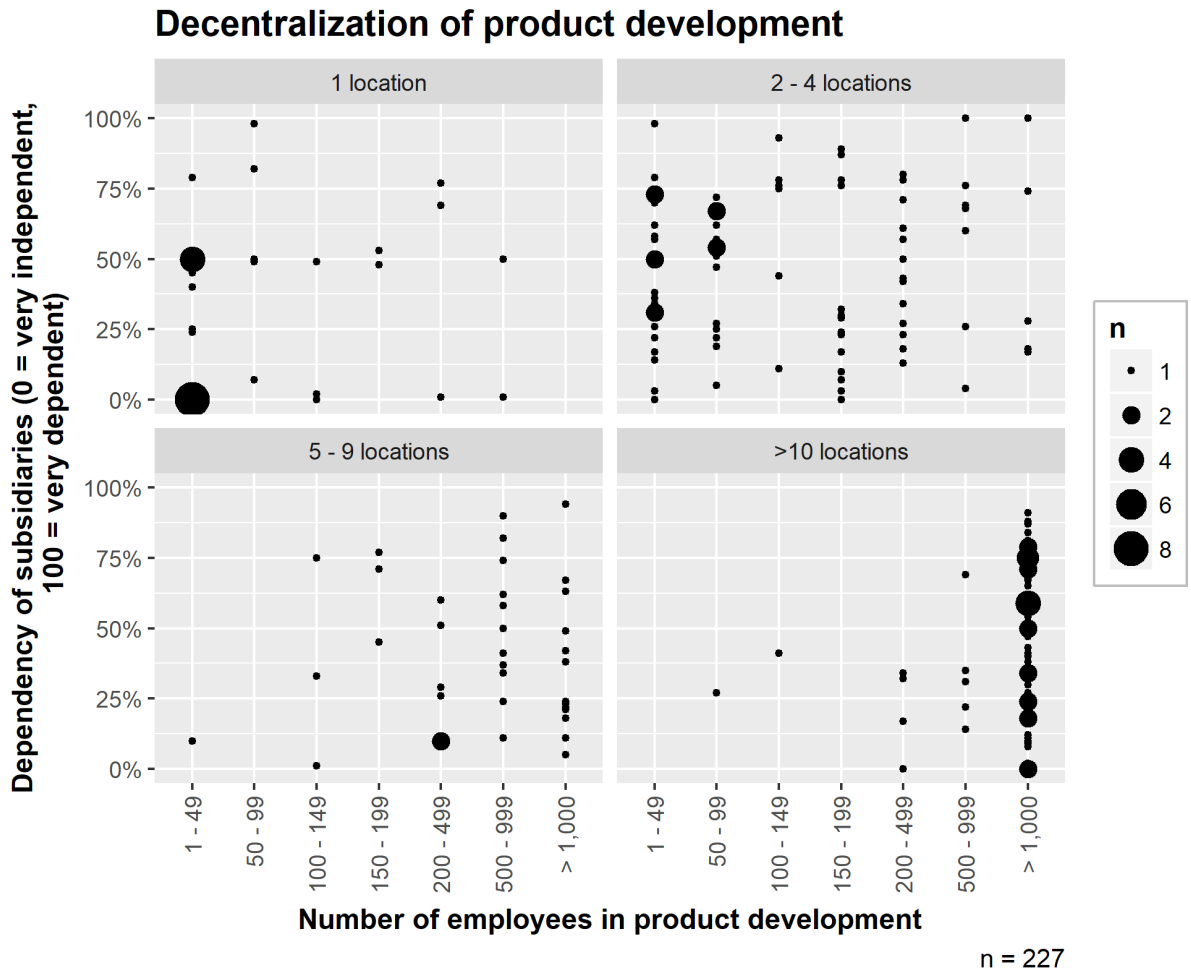


Figure 3.3.: Decentralization of product development of participating companies.

Description
Figure 3.3 shows the results from the perspective of the R&D organization (internal structure). The diagram summarizes 3 factors - number of product developers, number of development sites and dependency between the sites. The circles' diameter represent the number of answers for the particular combination.

Key learnings
- A large proportion of participants stem from organizations with more than 1,000 developers and more than 10 development sites.
- The development sites are strongly interdependent.
- The overall intrinsic complexity of product development is surprisingly strong.
- "Distributed product development" affects most of the participants.

Interpretation
- Agile development reaches from the original level of the individual team or domain level to the management of the entire development organization.
- There is a need to deduce ways of scaling the agile development concept.
- Cross-sectional and cross-locational problems have to be considered in the interpretation of subsequent diagrams.
- The internal complexity of the organization could affect the experience with the application of the agile development concept.

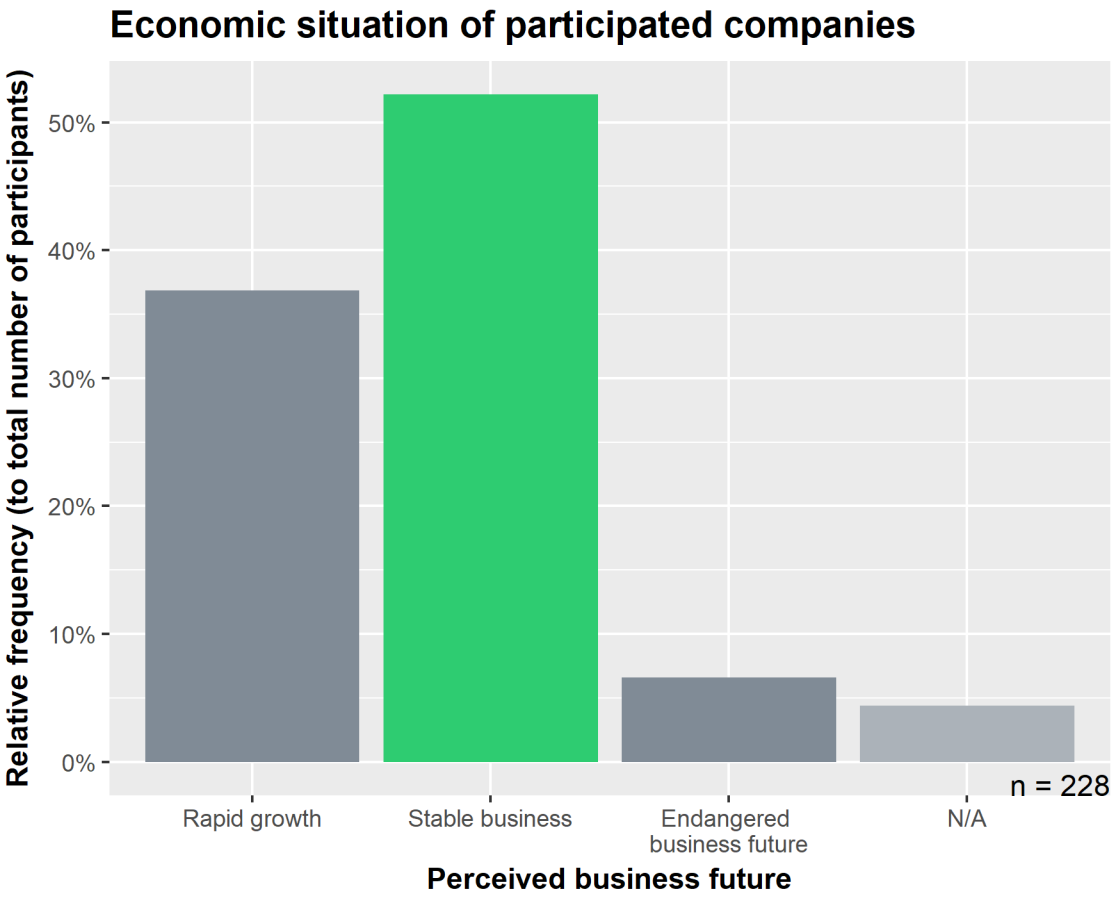


Figure 3.4.: Economic situation of participated companies.

Description
The assessment of the current situation of the companies is shown in Figure 3.4. The frequency of answers is divided into four categories: growth, stability, threat and no information.

Key learnings
- More than 50% of participants come from a stable economic environment.
- 38% of participants are in a situation of rapid growth.
- Less than 10% worry about their business.

Interpretation
- Attractiveness of agile product development for companies with difficult business situations is low, otherwise the survey would have attracted more participants from this category.
- Participants act from a situation of stable or growing business.
- Agile development is not associated with an acute emergency situation. It is not reported as a "lifeline" in a difficult situation. It is more of a future option for economic prosperity or for growth acceleration.
- The situations "stable" or "growing" could influence the effectiveness assessment of agile development, as improvements in satisfactory situations are often perceived rather weaker. The potential for a big step forward is subjectively smaller.

Interim Conclusions

Participants and their companies cover a wide range of manufacturing industries both in terms of size of companies and organization of development activities. In this respect, the study can be considered representative. At the same time, it can also be shown that the interest in agile development is large regardless of the size of the company and the organizational structure. Obviously, several challenges such as digitization, technological change and supply chains, especially in information and communication technology industries, are perceived as challenges where known methods of project management reach their limits and no longer contribute sufficiently to problem solving.

3.2. Product-related Questions

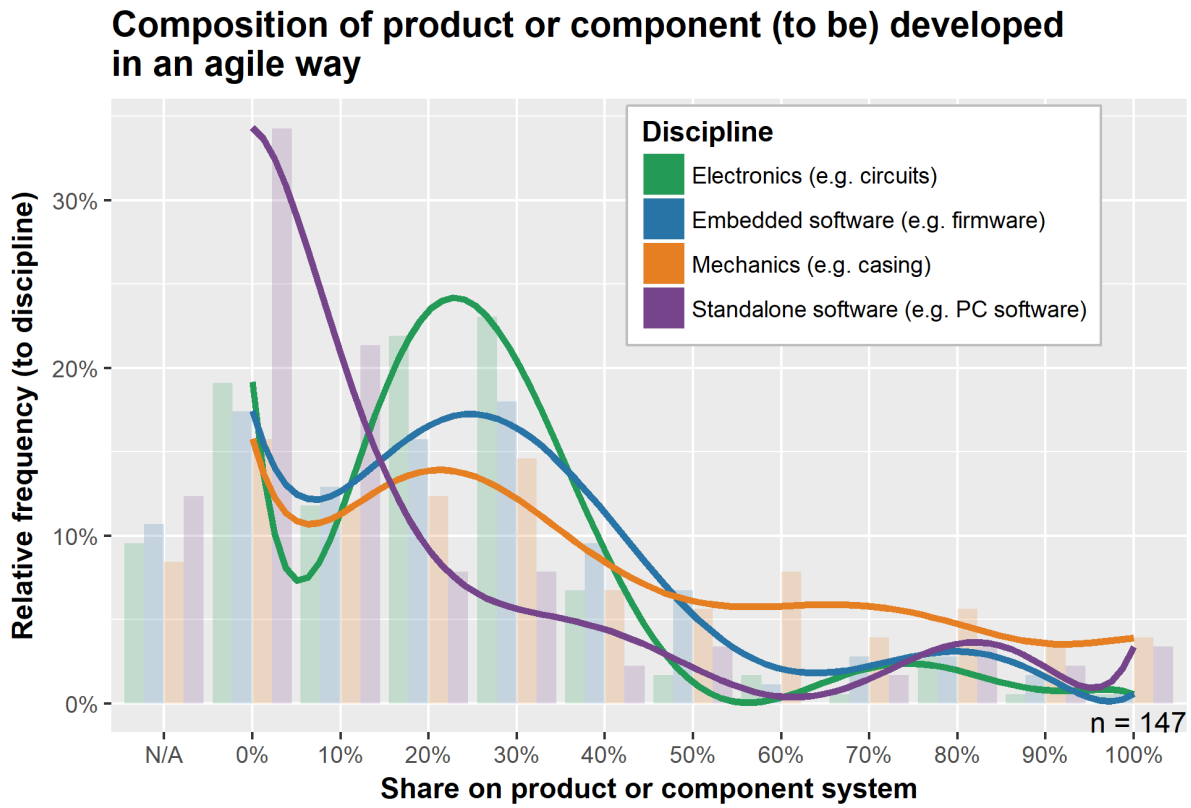


Figure 3.5.: Composition of product or component that is or shall be developed in agile manner.

Description
Single subject domains were used to characterize the product structure that participants are working on (Figure 3.5). The estimation of the proportion was left to the subjective impression of the participants. The values are not automatically equated with added value or expense proportions.

Key learnings
- Most products are mechatronic in nature.
- The contributions of the domains are distributed around a 1/3 ratio.
- Standalone software is poorly represented.
- Classic "one-dimensional" products are barely recognizable.
- Interdisciplinary cooperation is common.
- Physical problems are relevant for the participants.

Interpretation
- The participants experience is clearly based on the "Interdisciplinary Product Development" or domain overarching development.
- Today, agile development is not a methodology exclusively for software development.
- The need of electronic and software components is probably the driver that agile development has migrated and adapted into the practice of physical product development.
- The represented internal product structure is probably another driver of problem and solution complexity (see number of development sites), which promotes the interest in agile development methods.

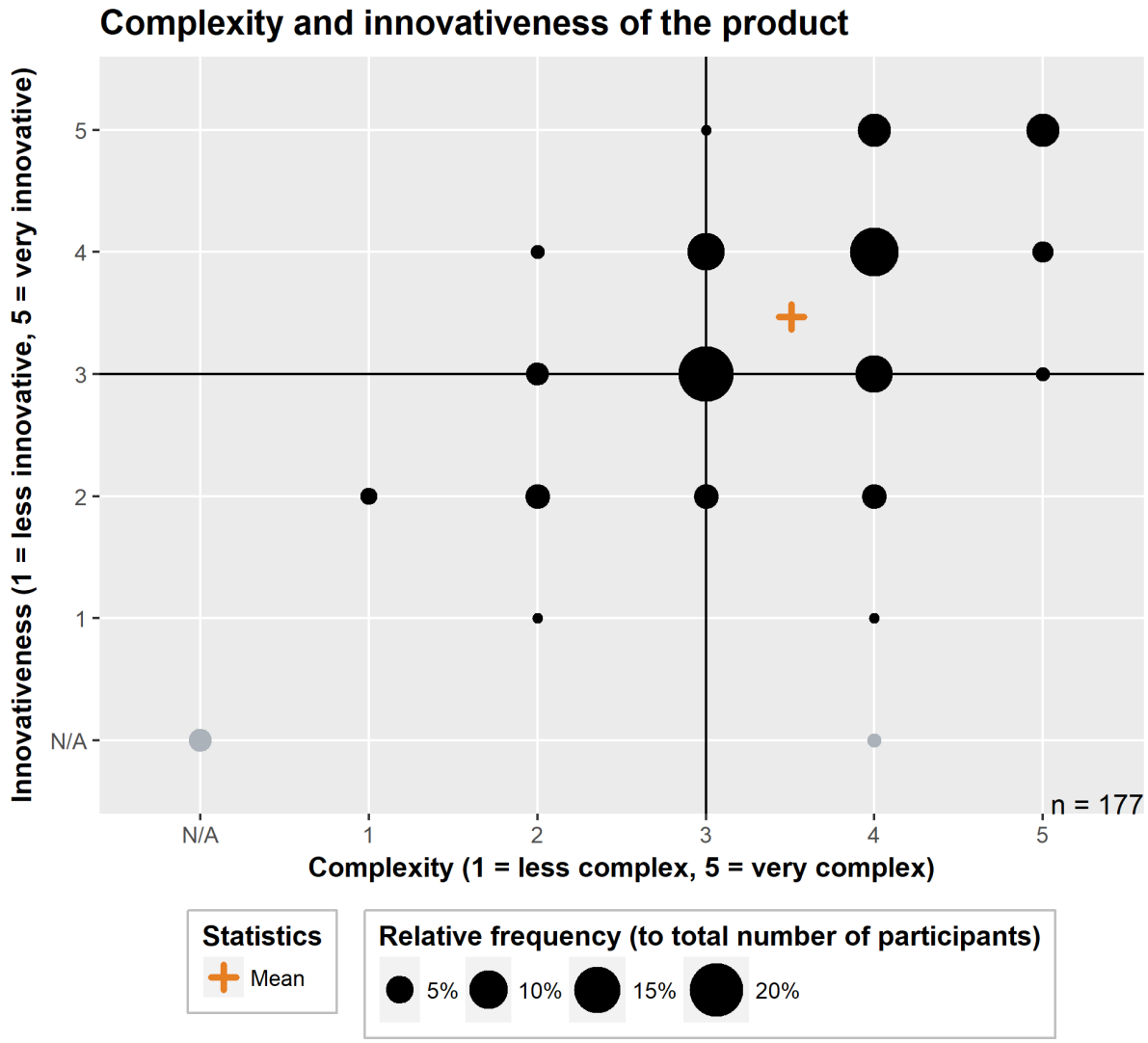


Figure 3.6.: Complexity and innovativeness of product that is or shall be developed in an agile manner.

Description

Figure 3.6 summarizes the answers to the question, which kinds of uncertainty the product development of the participants has to deal with. The parameters used were the assessment of the innovation challenge and the task complexity. The orange cross indicates the balance point, the mean on both dimensions. The grade 3 as the scales' medium splits Figure 3.6 in 4 quadrants.

Key learnings

- The balance point shows that the uncertainty stems from both innovative and complex tasks.
- 65% of the answers refer to very complex and innovative projects (upper right quadrant).
- The symmetry of the diagonal dividing line implies that the sense of innovation and complexity are closely linked.

Interpretation

- The positioning of the participants in the complex and innovative task field corresponds very well with the recommended field of application of the agile development concept.

Coping with high uncertainty is the goal of the agile development (see agile Manifesto).

- From the perspective of "perceived uncertainty", the study is in the desired field of experience.
- The need for novel methodological approaches is understandable from the participants' situation.
- There is a basic understanding for the appropriate context in which agile development is beneficial.
- It has to be considered that the scale was used subjectively, which can lead to a bias regarding an objective and factual level of "innovativeness".

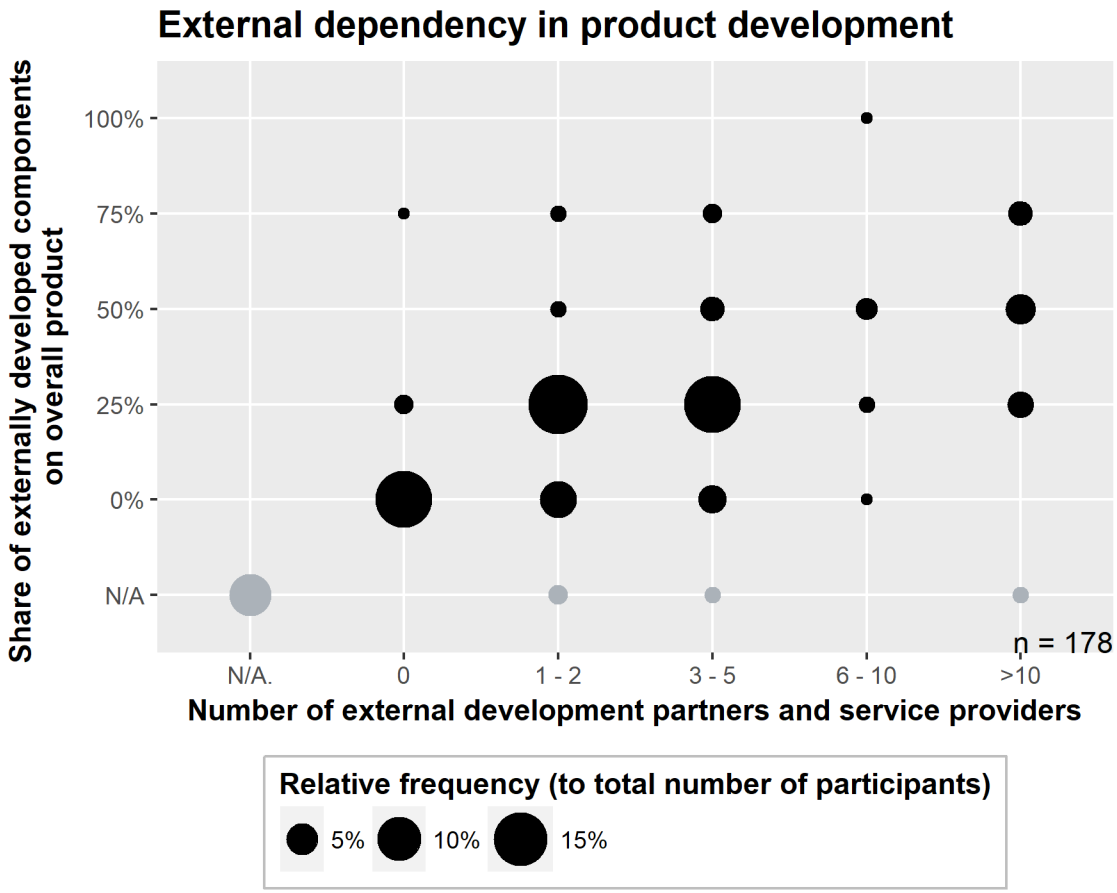


Figure 3.7.: External dependency in product development.

Description
Figure 3.7 shows the situation of product development from the perspective of external suppliers for product development. The proportion of externally developed product components over the number of development partners involved is shown. The circles represent the relative number of entries in percent to the total number.

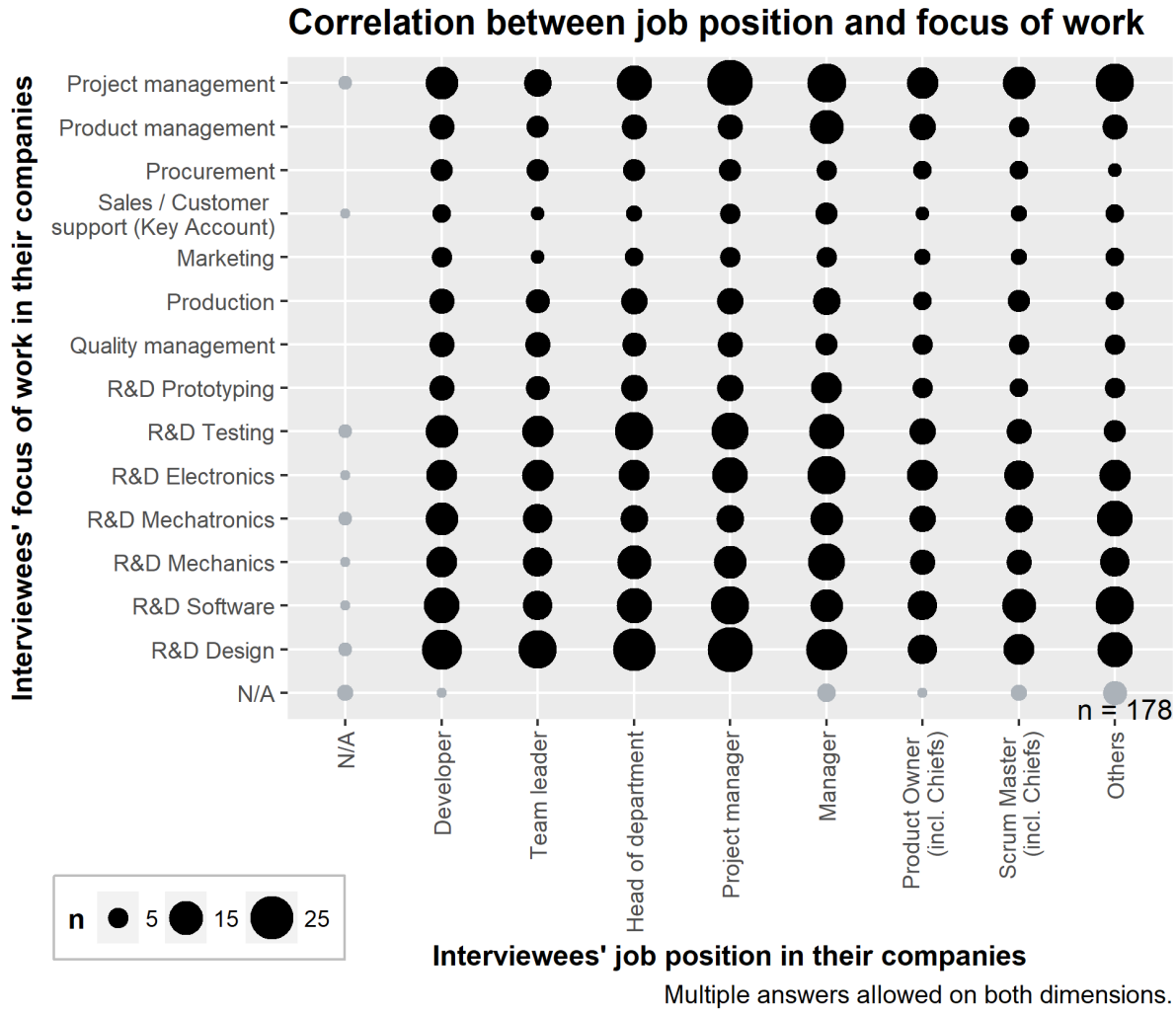
Key learnings
- Only 15% of the participants stated that neither external partners are involved nor components are developed externally.
- Approx. 85% of the participants operate a network in product development and obtains external component knowledge.
- The mean on the externally developed product share is about 25%.

Interpretation
- The results basically show the reality of externally networked product development. Product development obviously no longer takes place alone in the company - cooperation with contract partners is required.
- However, the focus with values of 1-2 of up to 5 partners and a component share of up to 25% is low compared to the value added of suppliers in product production.
- The dependence on externals could be challenging to the exploitation of the potential of agile development. There may be a need for scaling agile methods related to friction-less, smooth cooperation or collaboration that allows the integration of externals.

Interim Conclusions

The domain structure of the products addressed by the participants clarifies the frame of conditions under which the experiences are acquired. They are characterized by a high degree of complexity and innovativeness, by a high degree of interdisciplinarity and by a divided development approach due to company boundaries. All factors are well known drivers of complexity and uncertainty. On the one hand, the facts make it seem plausible that there is a need for new approaches in product development, because classical project management methods are reaching their limits. On the other hand, these framework conditions also make it necessary to think about adaptation and expandability of agile methods since the participants are confronted with factors that were not primarily in the focus of the original agile method landscapes.

3.3. Interviewee-related Questions



.

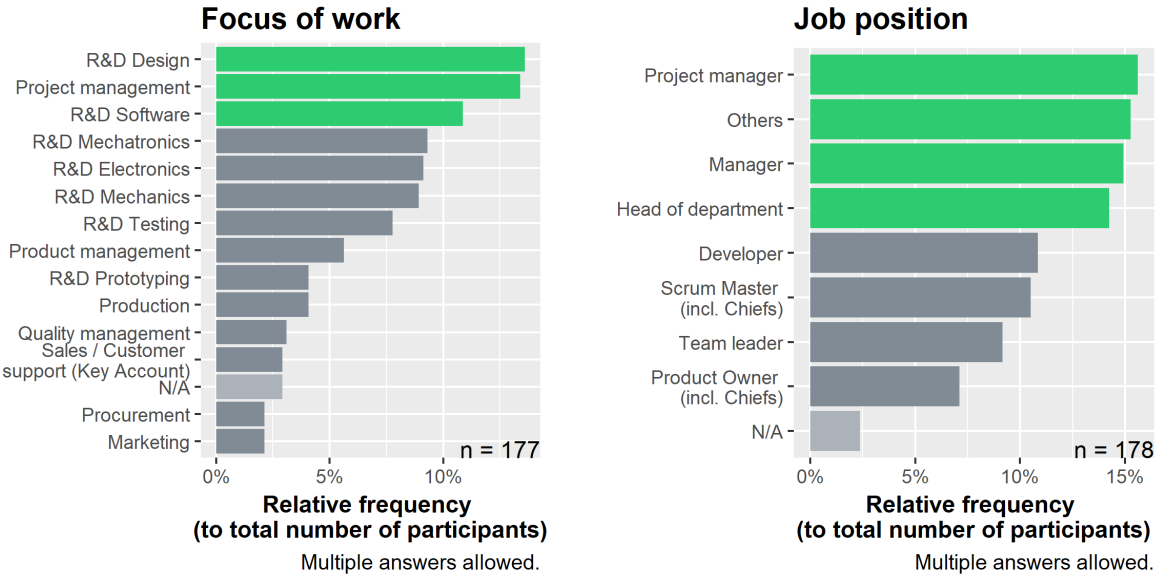


Figure 3.8.: Interviewees' background and distribution.

Description
At the first glance, the compilation in Figure 3.3 shows that the study results refer to a wide range of levels in hierarchy and job titles. The answer options were given. Multiple answers were allowed.

Key learnings
- The sample focus is clearly on activities and positions from development and development-related project work.
- Areas outside the R&D context are represented but smaller.
- All fields of product development work are represented in the sample.
- Classic development functions (domains) predominate. Production, purchasing, marketing, sales are also available.
- Most of the participants are Project Managers (16%), Managers (15%) or Heads of Department (14%).
- In the question about the position the category "Others" is noticeable. This can be explained by the diversity of in-house job titles.
- About 15% of participants already refer to typical Scrum roles such as Product Owner or Scrum Master.

Interpretation
- Approx. 80% of participants are not leaders by title (Managers or Heads of ..). The study captures the experience of operational project and product development activities.
- The existence of non-development focus of work (e.g. customer support, production) points to the overarching interest in agile development, even outside the classical development departments.

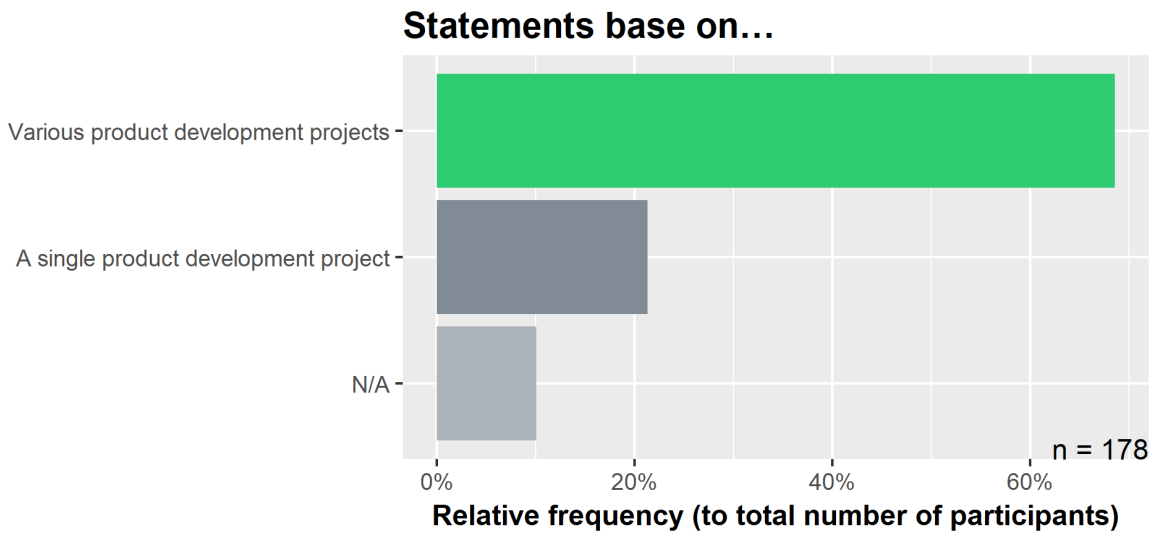


Figure 3.9.: Participants' scope of experience.

Description
Figure 3.9 breaks down the participants according to their experience in project work. The answer categories were given. Multiple answers were not allowed.

Key learnings
- 68% of the participants stated that they refer to experiences from several projects.
- Only 20% of responses relate to experienced gather by a single development project.

Interpretation
- The high proportion of multi-project knowledge indicates that most of the participants are involved in multi-project environments.
- The application of agile development is not a new territory for any of the participants. This again fits well with the reported proportions of Scrum Masters and Product Owners (see Figure 3.3).
- The representative relevance of the study is again strengthened by the multitude and the associated span of operational project experience.
- 5% of N/A answers can be reducible to non-project organizations (very small companies).

Interim Conclusions

The data of this section show that the study is based on a balanced cross-section regarding the hierarchical job levels. The results should therefore not be biased by a pure "management view", which minimizes the risk of reporting on "success stories". At the same time, the answers show that agile development has to include non-development domains in its application. However, the agile methods are not designed for that because they cover the core of technical development and project management.

The results also show that the response of the participants reflect predominantly broad levels of knowledge and experience. As a consequence, the expectation to create a representative cross-sectional study increases and should support the overall validity of the findings.

Part III.

Agile Development of Physical Products

4. Understanding of Agile Development

To interpret the expectations on agile hardware development, it seems essential to understand what the interviewees mean when they talk about agile development. Thus, the survey included questions that test the connotation and conception of agility as well as some closely linked aspects.

For that reason, the study firstly analyzes the understanding of agility by investigating adjective associations. In this way, terms were identified that can be used as synonyms or boundaries of agility.

Knowing which ability practitioners associate with agility, the study secondly questions the targeted field of application in development projects and tries to outline the meaning of working in an agile manner in development. The former refers to the difference between operative and administrative project work. For the latter, the author team created hypotheses based on often referred fundamentals to separate truths from myths.

To understand the range of agility towards the customer, the study thirdly included also questions concerning the sources of product definitions. The author team was keen on finding out, if companies base their product property definition on external (e.g. end user) or internal (e.g. sales department) customers when they develop it in an agile manner.

Fourthly, the study investigates the understanding of teamwork since the type of teamwork makes a difference between traditional and agile development. For that, cooperation- and collaboration-related questions have been asked.

Finally, the study analyzes what companies use to achieve agility in hardware development. The study tested so-called agile methods and practices. To explain the interdependencies between these terms., the analysis of the understanding of agile development concludes with testing a hypothetical framework.

4.1. Connotations of Agile Development

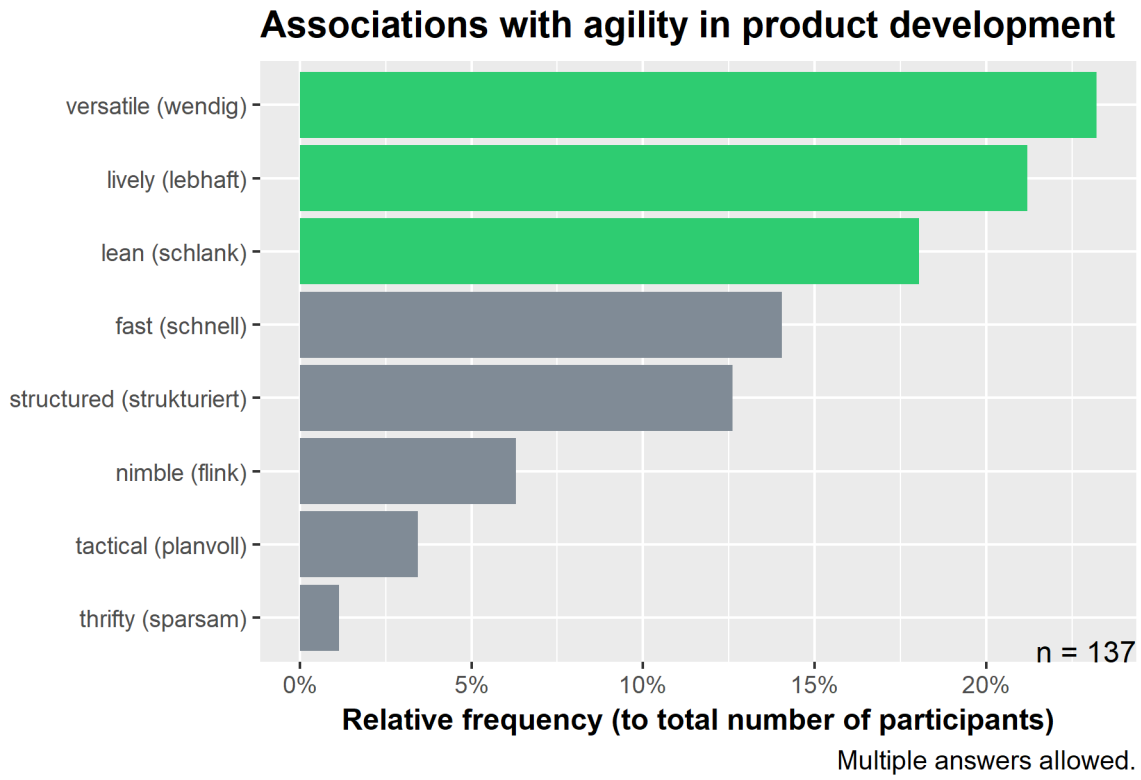


Figure 4.1.: Participants' associations with agile development of physical products.

Description
As translations might contain fuzziness, the German terms assessed are included in brackets. The translations fit to German terms as close as possible to the best of the authors' knowledge.

Versatile, lively and lean are associated with agility in product development mostly. On average, participants see only a few similarities to tactical and nimble. Thrifty might not be part of agile development at all. Although, agile development is often called chaotic, the adjective structured reaches a relatively high rank.

Key learnings
- Agility in development refers primarily to versatile which describes the ability to change.
- Lively that contains social aspects is largely associated with agile development as well.
- Some overlaps or confusions exist with lean development as lean is among the Top 3.

Interpretation
- It might be possible that participants do not differentiate between fast in terms of short time-to-market or fast in terms of immediate response to change.
- Being lean, but not thrifty is somehow contradicting.
- A unambiguous distinction between lean and agile hardware development, and their expected effects do not seem to be possible.

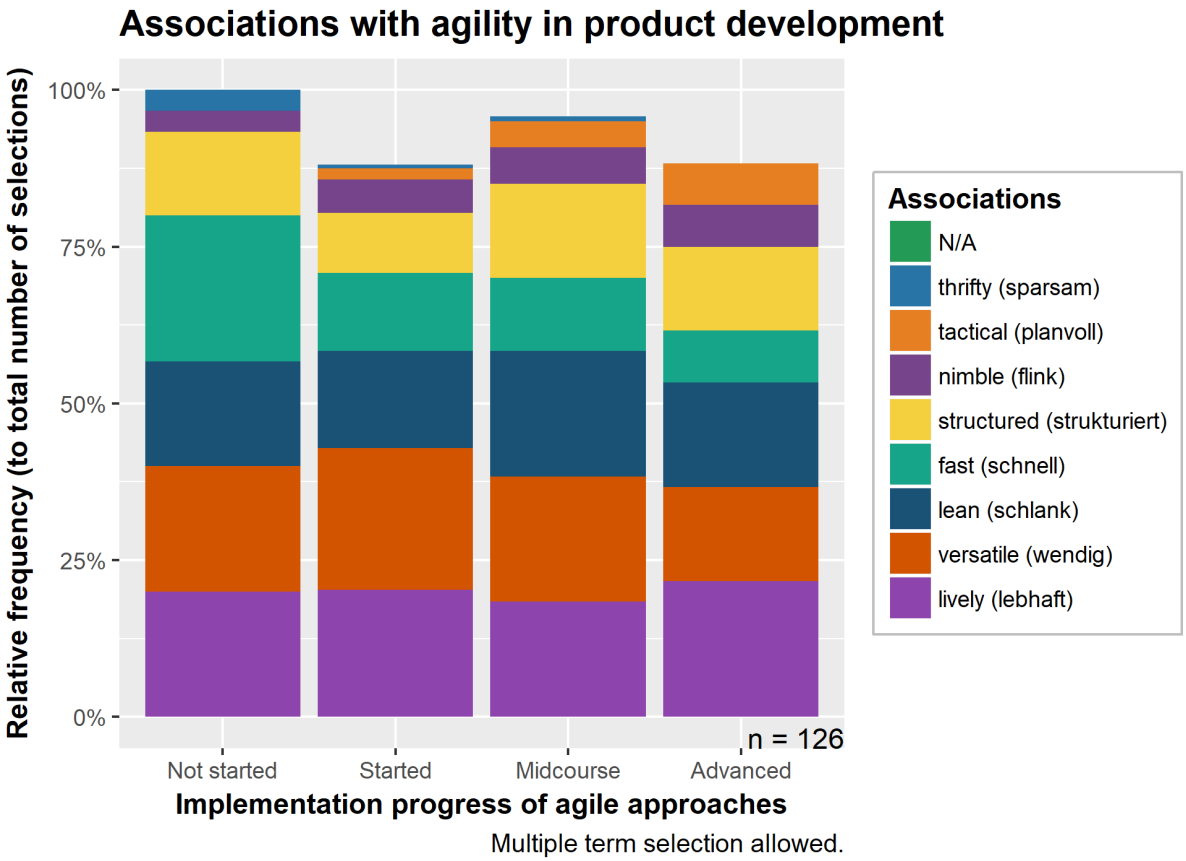


Figure 4.2.: Participants' associations with agile development of physical products correlated with the implementation progress.

Description
As translations might contain fuzziness, the German terms that were rated are included in brackets. The remaining part to 100% is caused by the fact that participants were allowed to choose 3 associations at maximum, but some chose only 1 or 2 terms.

Analyzing the connotation in correlation to the interviewees' experience reveals that shortening time-to-market ("fast") diminishes in importance with rising experience. In contrast, with increasing experience in agile hardware development, agility is more and more associated with "tactical", "structured" and "nimble". Associations like "lively", "versatile" and "lean" are stable, they do not correlate with the implementation progress a lot. Thrifty has only been chosen by those who have not yet started.

Key learnings
- The longer a company is engaged in agile development, the more it turns out that agility is tactical and not fast.
- Agile development has nothing to do with thriftiness.
- Companies that have not started yet associate agility in product development predominantly with "fast" (23%), "versatile" (20%) and "lively" (20%). However, advanced companies changed their opinion concerning "fast" (8%) drastically.

Interpretation
- Obviously, there is a misunderstanding between expectations and real improvements through agility especially when it comes to development lead times.

- It is striking that agility is less linked to "tactical". Due to the fact that agile development does not rely on a predefined project plan, planning as a continuous project activity seems to be essential, though.
- It remains unclear, if "creative" opposes "tactical".

Interim Conclusions

On the one hand, studying associated adjectives reveals a clear connotation of agile development in terms of flexibility in development. The ability to change rapidly is at the very core of agile development. It addresses challenges in reducing development risks that become more manageable through agility.

On the other hand, inflated expectations obviously exist when it comes to development lead times. Shortening time-to-market can be a positive side effect, but can not be considered as the actual motivation of agile development as it turns out in the course of this study, for instance, in Figure 6.2. Frustrations seems to be unavoidable when initial expectations of beginners prove to be unrealistic when they further progress in their implementation (see also Schmidt, Weiss, and Paetzold (2018)). In turn, this can lead to a general depreciation of agile development.

Besides that, an unambiguous distinction between agile development and other approaches such as lean development seems to be difficult. Depending on the companies' goals and contexts, approaches other than agile development may be more suitable in order to achieve them effectively and efficiently. There is a need for action to obtain information about potentials and limitations of those approaches. The following chapters shed light on these aspects for agile hardware development, but neglects comparisons to other development methods.

4.2. Conceptions of the Underlying Fundamentals

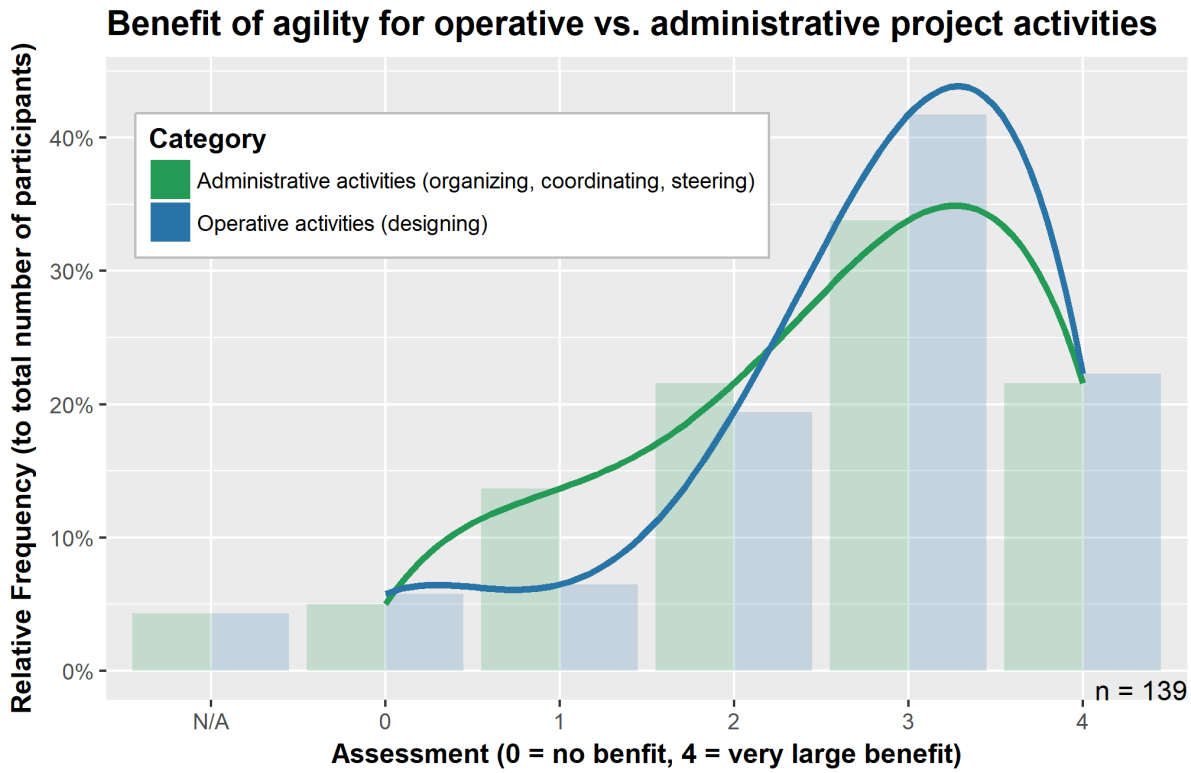


Figure 4.3.: Benefit of agility for operative vs. administrative project activities.

Description
It becomes apparent that agility has a benefit for both administrative and operative project activities. While the same number of interviewees answered the question on both categories, the administrative activities graph is more flat and receives more votes on lower benefit assessments. Concerning operative activities, votes that are missing on low benefit seem to be added on high benefit. Operative activities have a higher peak and have a higher mean value compared to administrative activities.

Key learnings
- Agile development includes both operative and administrative project aspects.
- Obviously, both operative and administrative project activities benefit from agility (operative a bit more than administrative activities).
- Agile development is not only about operative nor administrative project work. Separation (as in traditional product development: project management team vs. engineering team) does not seem to be possible.

Interpretation
- Although traditional development provides a wide variety of methods to solve concrete technical problems, companies see large additional benefits in agile development for operative project work.
- The benefit of agile development might be caused by the fact that it addresses rising systems complexity and uncertainties that also lead to higher administrative effort. Traditional development approaches seem to fall short here.

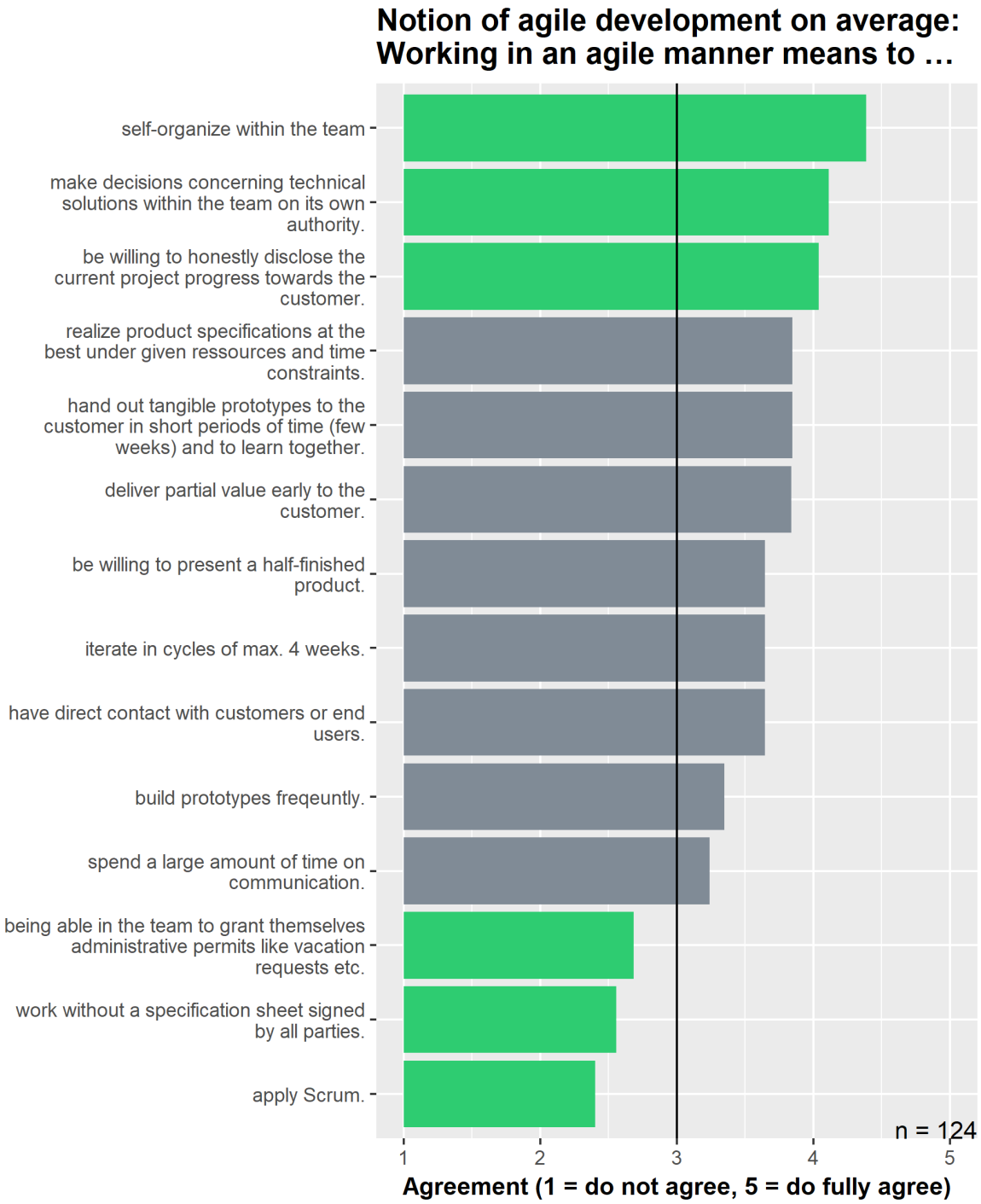


Figure 4.4.: Hypotheses about the notion of working in an agile manner.

Description
To detail the conceptual meaning of working in an agile manner in physical product development, 14 hypotheses have been created and tested in the survey. Interviewees rated their agreement on a scale between 1 (do not agree) and 5 (do fully agree).

On average, participants agree on most of the hypotheses as the overall mean displayed is above 3 in most cases. On the one hand, self-organization, team authority about technical issues and transparency towards customers are rated highest. On the other hand, interviewees do not relate agility in development to the application of Scrum, the disregard of signed specification sheets

by all parties and the team authority to self-organize when it comes to administrative permits like vacation requests.

Key learnings

- Agile development is not just about applying Scrum.
- Self-organization, team authority in technical issues, and transparency are very closely associated with agile development.
- Although prototyping seems to be linked to agile development, it is less important than self-organization, team authority or transparency.

Interpretation

- Companies might still think in traditional ways when calling for self-organized teams, but the teams' responsibility is limited to technical issues only.
- It seems to be worthwhile to question what self-organization of agile development teams does include and what not. Why should self-organization not also cover e.g. vacation requests?
- Most interviewees do not question the existence of product specification sheets (Lasten-/Pflichtenhefte). It remains open, if classical specification sheets are appropriate under dynamic circumstances that agility addresses.
- Agile development might be limited to internal affairs. Integrating real end users, sponsors, maintainers etc. who are external (from companies) in most cases does not seem to happen often. Direct customer interaction and integration, however, proves agile development literature to be very powerful.

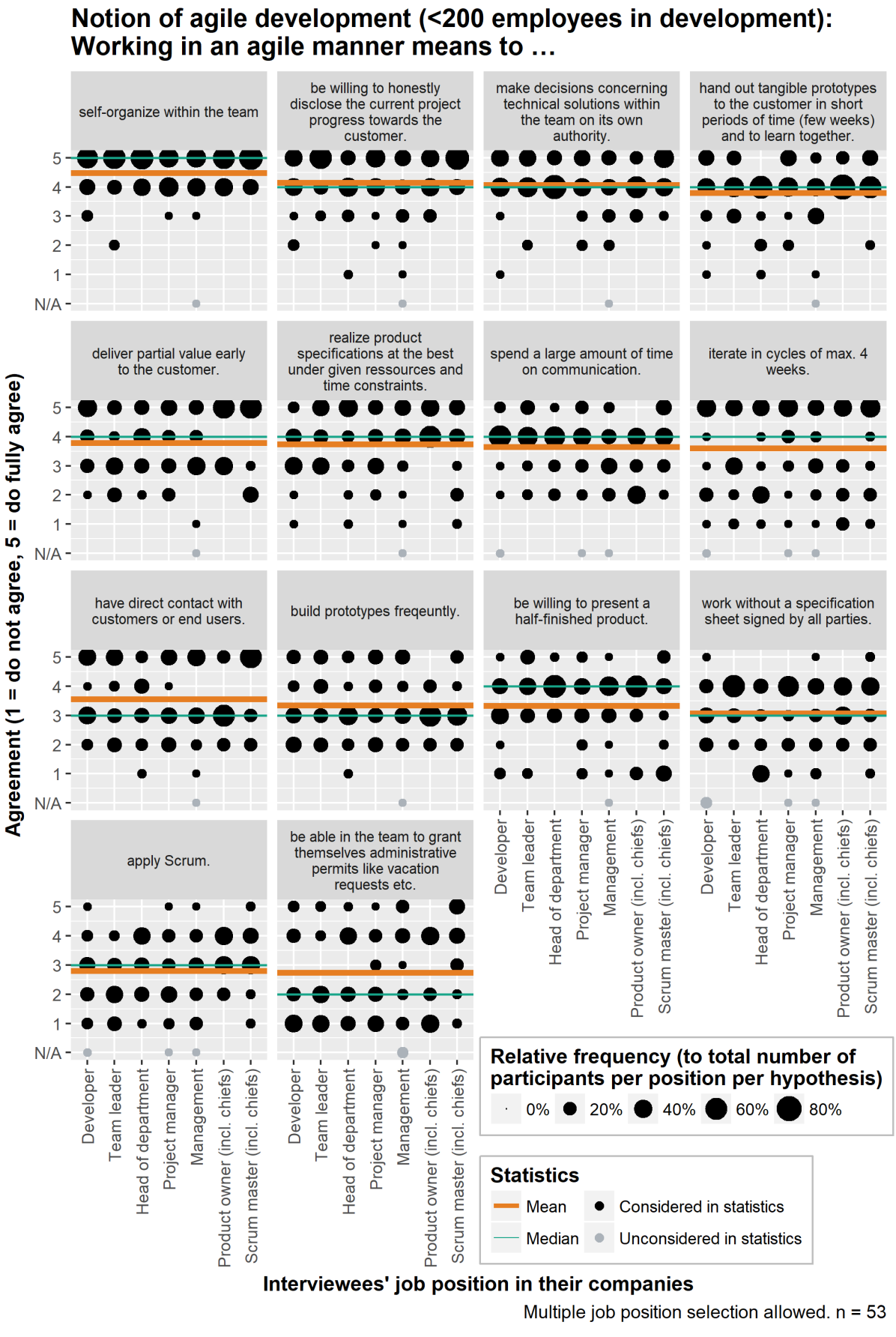


Figure 4.5.: Notion of agile development in companies having less than 200 employees in their product development department sorted by decreasing means.

Description
Data visible is a subset only containing small and medium-sized companies with less than 200 employees (n=53). Filled circles in the diagram represent interviewees' answers. Their size is normalized to the total number of interviewees having the same job position per hypothesis. Thus, the sum of each column (job position) of each hypothesis box equals 100%. The diagram depicts the deviations for each hypothesis concerning different perspectives.

While the deviations are small especially for hypotheses with high means, they are quite large for those showing low means. However, deviations differ among job positions. While, for instance, Scrum masters certainly agree on "Working in an agile manner means to be willing to honestly disclose the current project progress towards the customer", the managers' answers spread largely. Managers do not associate agility in development with transparency towards customers like Scrum masters do. To name another example, head of departments associate agility more certainly with the willingness to present half-finished products than developers.

Key learnings
- The notion of agile development in companies having less than 200 employees in development deviates a lot in many aspects.
- Self-organization, transparency towards customers and team authority concerning technical issues is associated most with agile development (mean > 4).
- Especially when it comes to administrative permits, Scrum and product specification sheets, opinions spread a lot. These hypotheses in particular do not seem to be job position-dependent.

Interpretation
- Agility in hardware development is not a mature concept yet. Large potentials for interpretation how to bring the concept from software development to hardware development seem to exist.
- What agility means to companies with less than 200 employees in development depends strongly on who is being questioned.

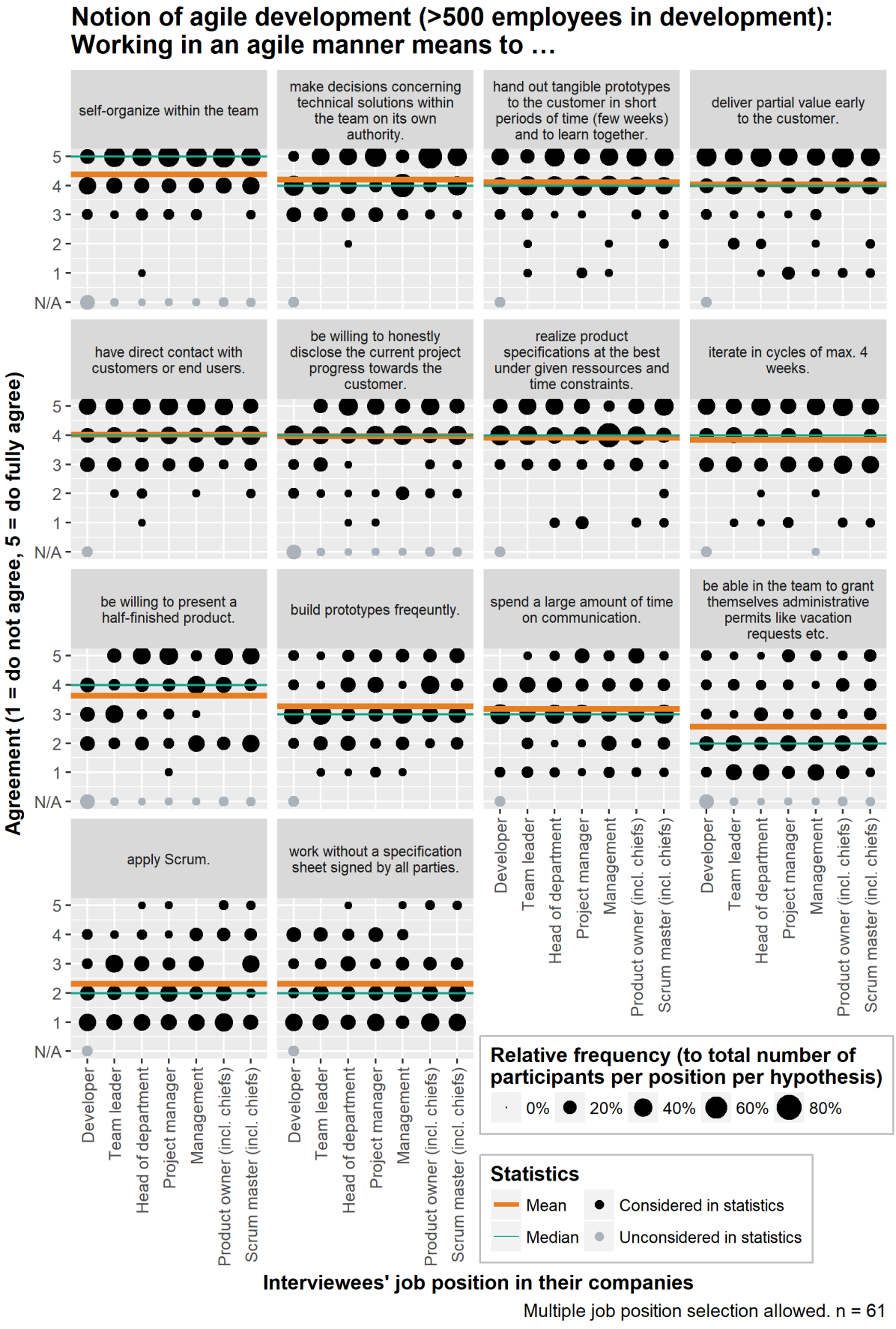


Figure 4.6.: Notion of agile development in companies having more than 500 employees in their product development department sorted by decreasing means.

Description
Data visible is a subset only containing large companies with more than 500 employees in development departments (n=61). Filled circles in the diagram represent interviewees' answers. Their size is normalized to the total number of interviewees having the same job position per hypothesis. Thus, the sum of each column (job position) of each hypothesis box equals 100%. The diagram depicts the deviations for each hypothesis concerning different perspectives.

While the deviations are small especially for hypotheses with high means, they are quite large for those that have low means. However, deviations differ among job positions. While, for instance, developers rather do not associate agility in development with the willingness to present half-finished products, project managers are very uncertain, but agree to a rather high level on average. To name another example, "Working in an agile manner means to make decisions concerning technical solutions with the team on its own authority." is rated with 5 by more than 80% of participated product owners, while developers agree with it to less than 4 on average.

Key learnings
- The notion of agile development in companies having more than 500 employees in development deviates a lot in many aspects.
- Self-organization, team authority concerning technical issues, prototyping and learning jointly with customers, delivering early customer value, direct customer contact, discosing the current project progress is associated most with agile development (mean > 4).
- Especially when it comes to product specification sheets, Scrum and administrative permits, opinions spread a lot. These hypotheses in particular do not seem to be job position-dependent.

Interpretation
- Agility in hardware development is not a mature concept yet. Large potentials for interpretation how to bring the concept from software development to hardware development seem to exist.
- What agility means to companies with more than 500 employees in development depends strongly on who is being questioned.

Interim Conclusion

Comparing Figure 4.5 and Figure 4.6 shows that large companies have a more extreme notion of agile development. Self organization, decision making authority etc. is rated higher and administrative permits, Scrum and disregard of specification sheets is rated lower compared to the answers smaller companies have given.

Interestingly, for the hypothesis "Working in an agile manner means to be able in the team to grant themselves administrative permits like vacation requests etc." the survey reveals two contradicting groups of participants in small and mid-sized companies (<200 employees in development). While one claims that it is part of agility, the others disagree totally. On average, the mean is almost 3. This, however, is different for large companies since two distinct groups are not recognizable there.

Similar, but less deviation can be seen in hypotheses concerning direct customer contact, cycle time, letting the team grant administrative permits (like vacation requests etc.) and early customer value for small and medium-sized companies. For large companies, these groups are less obvious or distinct, but express in form of general deviations. Employees in large departments are more specialized and limited in their tasks while in small ones the variety of tasks and responsibilities for each developer is much wider.

Differences in "Working in an agile manner means to make decisions concerning technical solutions within the team on its own authority" between Figure 4.5 and Figure 4.6 points to different power and decision making processes. Companies with large development departments seem to be challenged with decentralizing decision making and flatting hierarchies.

Comparing Figure 4.5 and Figure 4.6 also reveals that max. cycle time of 4 weeks is recognized in large development departments, but less in small and medium-sized departments (due to higher deviations).

In conclusion, self-organization and decentralized power is obviously perceived as an approach to deal with complexity and uncertainty. As a consequence, especially organizational hierarchies in large companies are challenged. Calling for freedom in decision and action requires decentralization and alternative communication structures. There seems to exist a certain mismatch between agile methods and large organizations. Method adaption, effective scaling strategies or restructuring of companies' organizational structures are inevitable to become truly agile. Nevertheless, many cause and effect interdependencies are less known and imply a need for action.

4.3. Sources of Product Requirements

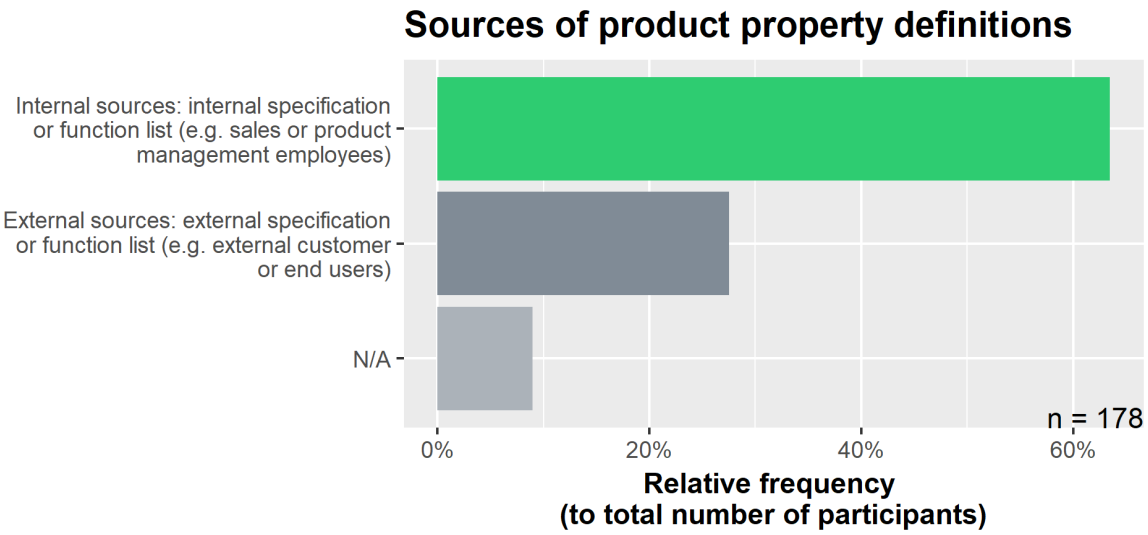


Figure 4.7.: Sources of product property definitions in participated companies.

Description
In general, product properties that describe what to develop (e.g. classical product requirements, functions or value descriptions) can originate from internal or external sources. When the development team receives those product specifications, for instance, from sales, product management or top management, it represents internal sources. Product properties are externally sourced when, for instance, external customers or actual end users submit them. Norms and standards can be both internal or external sources.

On average, the main source of product property definition seems to be internal ($> 60\%$). The external sources were only picked by a little less than 30% and about 10% of the interviewees chose the N/A option.

Key learnings
- 2 of 3 products to be developed or developed in an agile manner are specified internally.
- 1 of 3 products to be developed or developed in an agile manner are specified externally.

Interpretation
- The Manifesto of agile development calls for customer collaboration, customer satisfaction and customer value. It remains questionable, if product development addresses the right needs when relying on internal sources.
- The difference between consumer vs. capital goods as well as market- vs. technology-driven industries might matter because requirements are triggered differently.
- In some markets, external customers or end users could be not willing to have intensive collaboration with development teams which leads to customer representations (internal source).
- Potentially, employees could be not open to sense the right needs when the corporate culture or organization does not sensitize it - e.g. due to separated, maybe mutual exclusive goals and responsibilities of the departments (development, variant management, sales etc.).

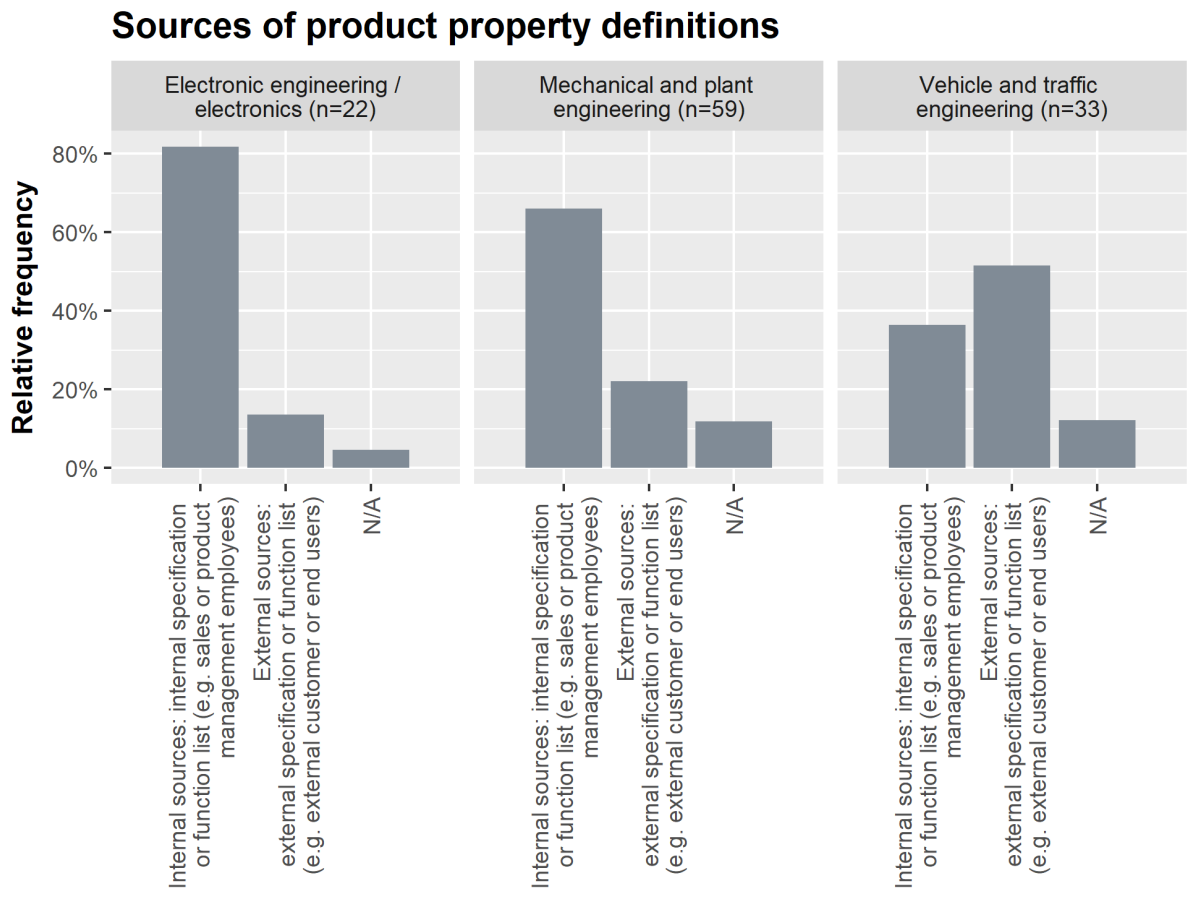


Figure 4.8.: Sources of product property definitions compared among industries.

Description
Although the total number of participants per industry is relatively small, the general tendency is obvious. While electronic engineering / electronics assembly, and mechanical and plant engineering specify their products internally, product properties in vehicle and traffic engineering industry are defined by external sources mostly.

Key learnings
- Sources of property definition are industry-dependent.
- Some industries (e.g. vehicle and traffic engineering) let externals define what to develop.

Interpretation
- According to the Manifesto of agile development, external product property definitions seem to be more conform with the agile mindset as it calls for customer collaboration, customer satisfaction and customer value.
- The risk to miss the market needs seems to be higher in some industries.
- In contrast to the electronic engineering industry, the vehicle and traffic engineering industry is a supplying industry predominantly. OEM's specify many requirements very detailed (external sources).
- It seems likely that market-driven industries base their product development on external sources, while technology-driven industries have mainly internal sources. However, less evidence is present in collected data.

Interim Conclusion

Most companies from Germany, Austria and Switzerland that participated in the survey rely on internal product property definition sources currently. They base their product specification on e.g. sales or product management departments that are in charge of scanning and reflecting the right market needs. This leads to internal customer representations. Since more than 60% of the participants fall in this category, agile development of physical products needs to consider that.

However, according to the Manifesto of agile development, external product property definitions seem to be more conform with the agile mindset as it calls for customer collaboration, customer satisfaction and customer value. Nevertheless, external sourcing can be challenging as it comes with a mental change for many stakeholders, e.g.:

- Customers could be not willing to be involved in development frequently - they just want to get the job done.
- Finding the right external person who should be allowed to specify product properties can be difficult especially in mass markets.
- Companies could try to separate the responsibility for the right product specification to specialized departments such as sales.
- Moreover, the way how customers and developers communicate and collaborate requires rethinking in agile hardware development as, for instance, the constraints of physicality limit the possibilities to build cheap prototypes rapidly. Handing out tangible prototypes to the customer in short periods of time, however, is considered one of the key fundamentals of agile hardware development (compare Figure 4.4).

Practically, the evaluation whether internal or external product property definition is more suitable for agile hardware development depends heavily on context aspects like consumer vs. capital goods, and technology push or market pull strategies. Product properties are triggered differently with various (strategic) intentions.

4.4. Understanding of Teamwork in Agile Development Projects

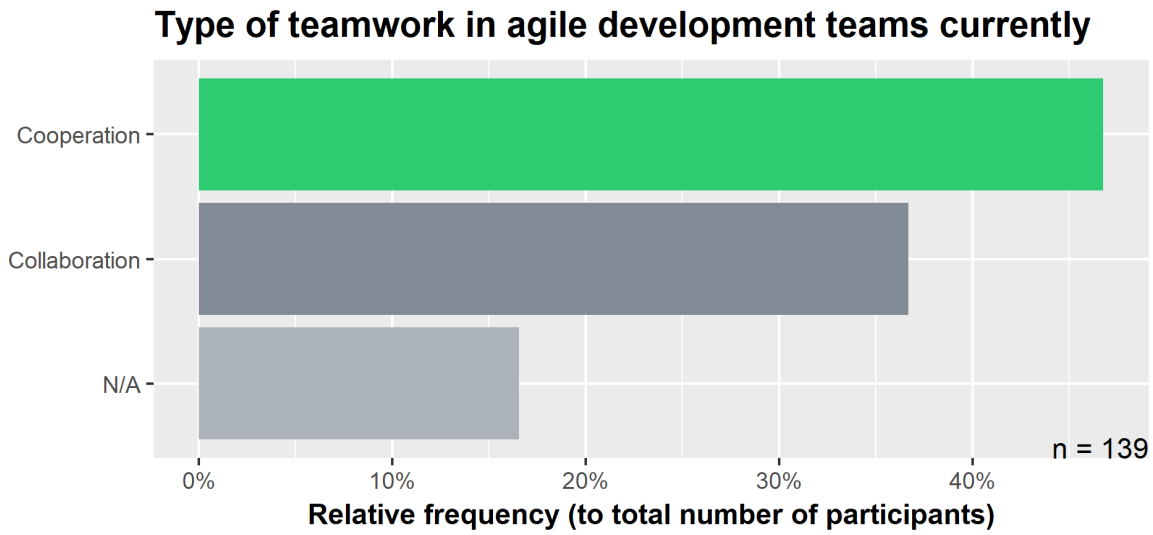


Figure 4.9.: Current type of teamwork in teams developing physical products in an agile manner.

Description
Following definitions of collaboration and cooperation were given in the survey.

- Collaboration: Solving project tasks simultaneously and jointly with other team members. Developing the final project result as a group. The achievements of each single team member cannot be distinguished usually.
- Cooperation: Solving project tasks in independent components which team members can accomplish alone and in parallel to each other. Team members deliver distinct parts. The sum of all parts forms the final project result.

More than 45% of the participants have a cooperative type of teamwork in agile development currently. Only 37% develop physical products in an collaborative manner. The N/A option has been used relatively often for that question.

Additional information: N/A has been chosen mainly by beginners (not yet started or just started). The further in implementation progress, the more equal becomes the current type of teamwork ("advanced" as much votes for collaboration as for cooperation while cooperation dominates for "started"). Compare Figure 7.9 for the implementation progress categories.

Key learnings

- Cooperation is currently the most established type of teamwork within agile development of physical products.
- However, collaborative working is almost as common as cooperative working in agile product development.

Interpretation

- Although the type of how to work with the customer is specified in the Manifesto (being collaboration), it does not describe the way how to work within the development team. Collaboration, however, seems to be too rare currently.
- Differentiation between the two concepts could be challenging for practitioners, e.g. because difference between collaboration and cooperation is not common knowledge or because a mix of both is present in real life teamwork (no dominating concept).

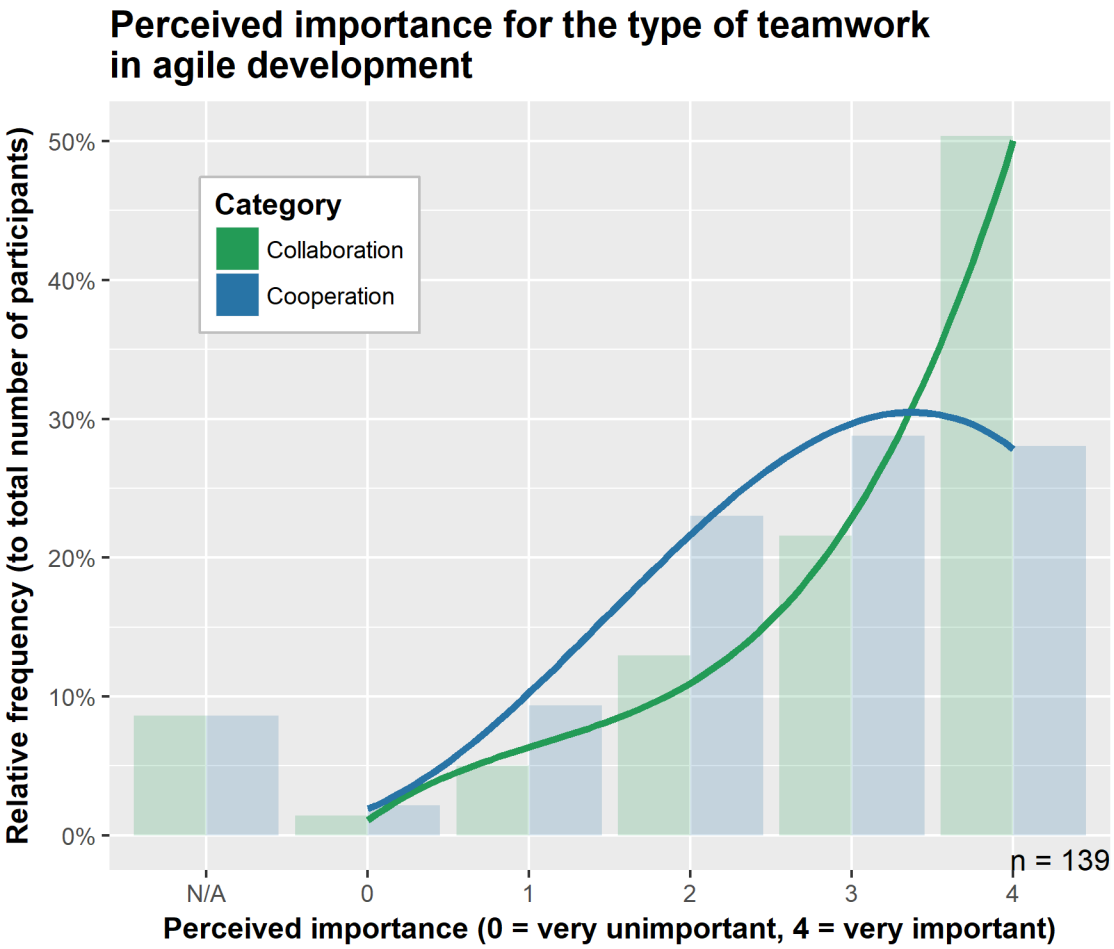


Figure 4.10.: Perceived importance for the type of teamwork in teams developing physical products in an agile manner.

Description
Both collaboration and cooperation graphs are left-skewed whereas collaboration increases progressively while cooperation reaches a maximum at about 3. Although the average assessment of cooperation (2.8) is smaller than the average of collaboration (3.4), both within the upper half of the scale.

Additional information: N/A has been chosen mainly by beginners (not yet started or just started). The curve shape of collaboration is independent from the implementation progress. When it comes to cooperation, the curve peak is at 3 for "started", but at 2 for "midcourse". Compare Figure 7.9 for the implementation progress categories.

Key learnings
- While collaborative teamwork is perceived more important for agile development, cooperation should not be neglected.
- In general, agile development is strongly linked to teamwork. Performing as a team in whichever kind of teamwork seems to be very important.

Interpretation
- Although the type of how to work with the customer is specified in the Manifesto (being collaboration), it does not describe the way how to work within the development team. Collaboration, however, seems to be more valuable for agile development teams.

Interim Conclusion

Teamwork is perceived as fundamental aspect of agile hardware development, which is also coherent with Figure 4.4. Especially collaboration in contrast to cooperation turns out to be highly important. However, not even companies which are experienced in agile hardware development have achieved their desired level of collaboration. Considering Figure 7.4, today's organizations or mindsets do not seem to be ready for collaboration.

4.5. Methods and Practices Used

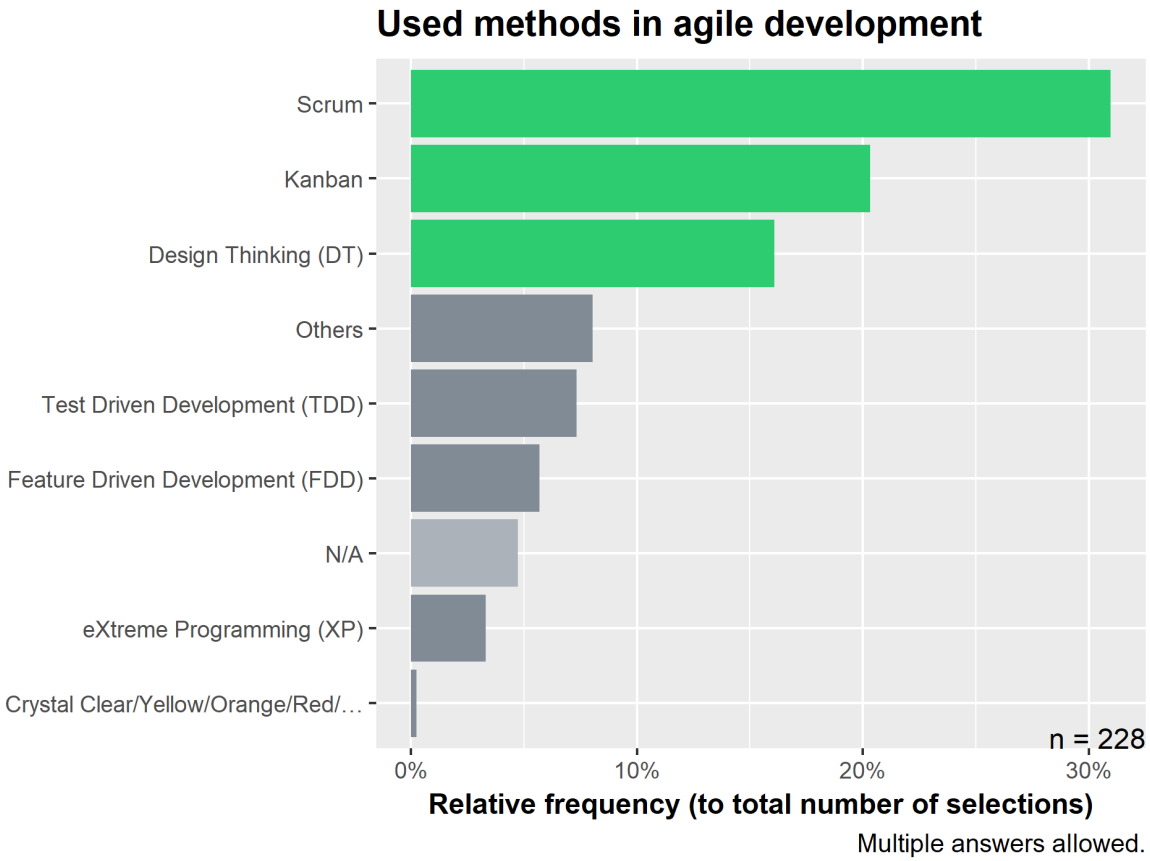


Figure 4.11.: Methods used in agile development of physical products by participated companies.

Description
Scrum, Kanban and Design Thinking are the most used methods in agile development of physical products. Crystal, eXtreme Programming, Feature Driven Development and Test Driven Development does not play a major role.

Additional information: "Others" has been chosen mainly by experienced participants (mid-course, advanced or completed in implementation).

Key learnings
- Scrum, Kanban and Design Thinking are the most popular methods in agile development of physical products.
- Although eXtreme Programming had a massive impact on agile software development, it plays almost no role in agile development of physical products.

Interpretation
- It remains questionable, if Scrum, Kanban and Design Thinking is used only because others use it or due to extensive marketing efforts of their creators.
- When practitioners talk about agile hardware development, they refer mostly to Scrum, Kanban and Design Thinking. This leads to the highest visibility of those methods in public.

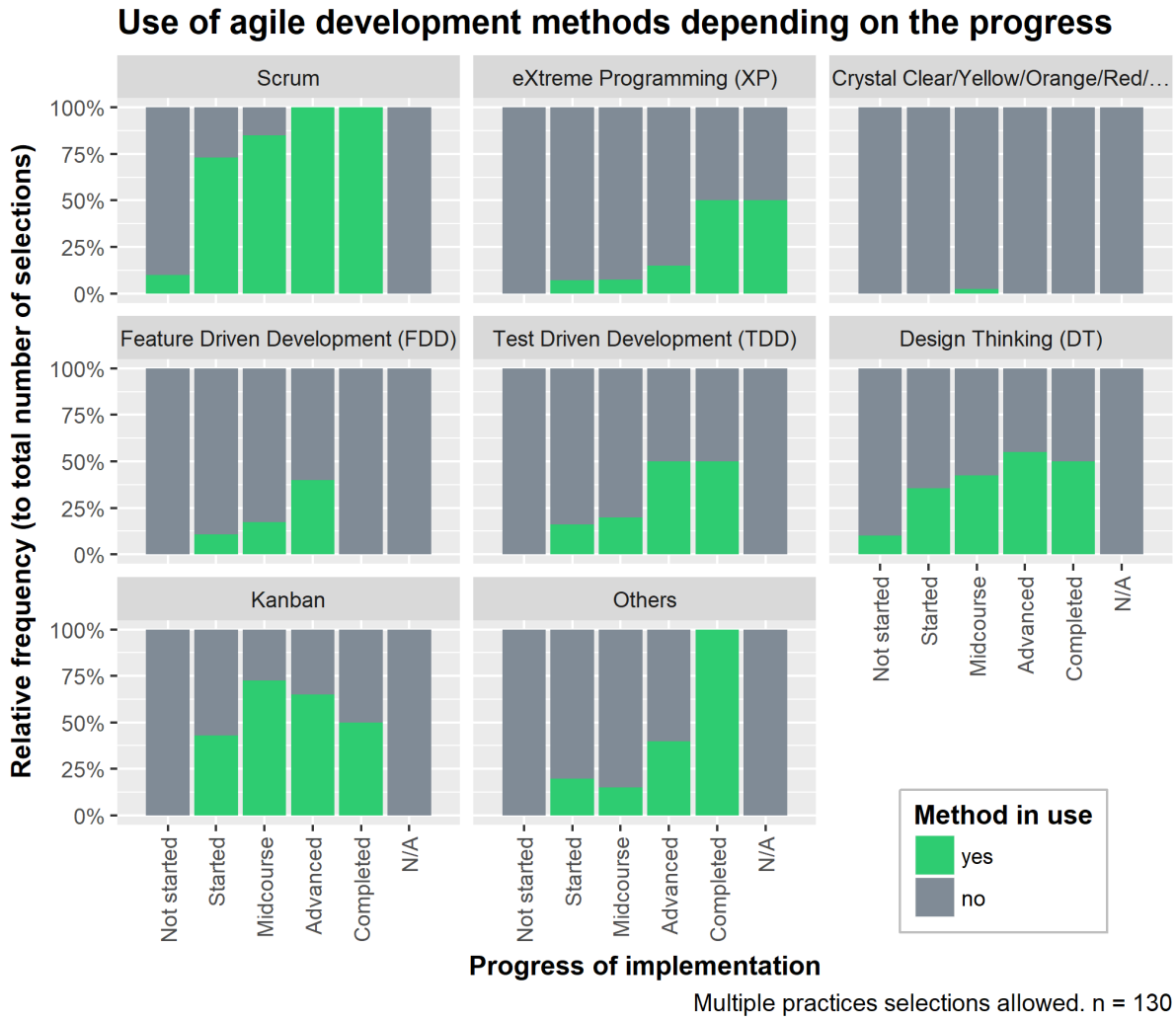


Figure 4.12.: Methods used in agile development of physical products in correlation with the implementation progress.

Description
Concerning methods in agile development, almost 75% of all partcipated companies that just started with the adoption begin with Scrum. Kanban seems to be implemented especially midcourse. Obviously, companies increase the variety of methods they use over time. However, Kanban is the only method investigated companies use less over time. Test and Feature Driven Development becomes interesting for companies when they are advanced.

Among the participants only two said that they completely implemented agile development and another two chose N/A. The "completed" and "N/A" bar is negligible and are shown only for the sake of completeness. See Figure 7.9 for the number of participants per progress category.

Key learnings
- Most companies start with Scrum.
- Kanban does not seem to fulfill the need of advanced agile development teams.

Interpretation
- Popular methods like Scrum, Kanban and Design Thinking are starters that get adapted to specific contexts.

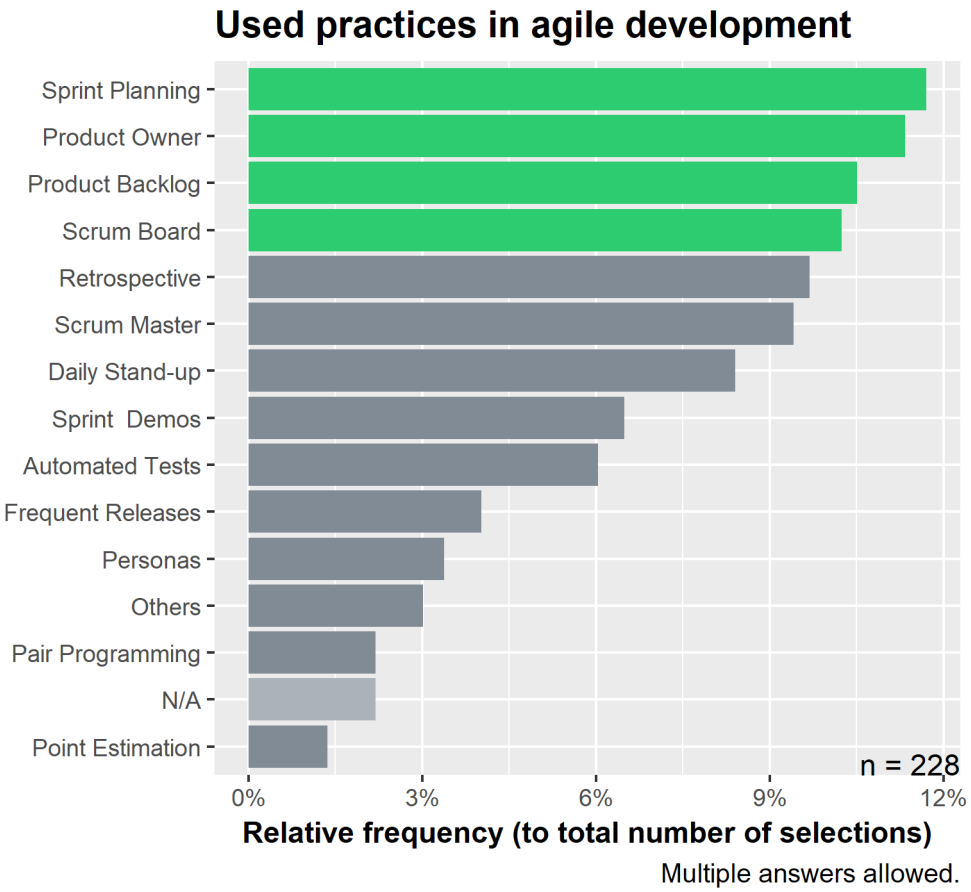


Figure 4.13.: Practices used in agile development of physical products by participated companies.

Description
On average, Scrum corresponding practices are used most. In particular, those are sprint planning, product owner, product backlog and Scrum Boards. However, pair programming which might be practicable in e.g. CAD, too, is used rarely.

Key learnings
- Practices corresponding to Scrum are used most.
- Practices supporting operative project work (such as pair programming, personas and automated tests) are chosen relatively rarely.

Interpretation
- It seems that especially project management-related aspects of Scrum are seen valuable and thus lead to frequent adoption in practice.
- Agile methods have overlapping practices.

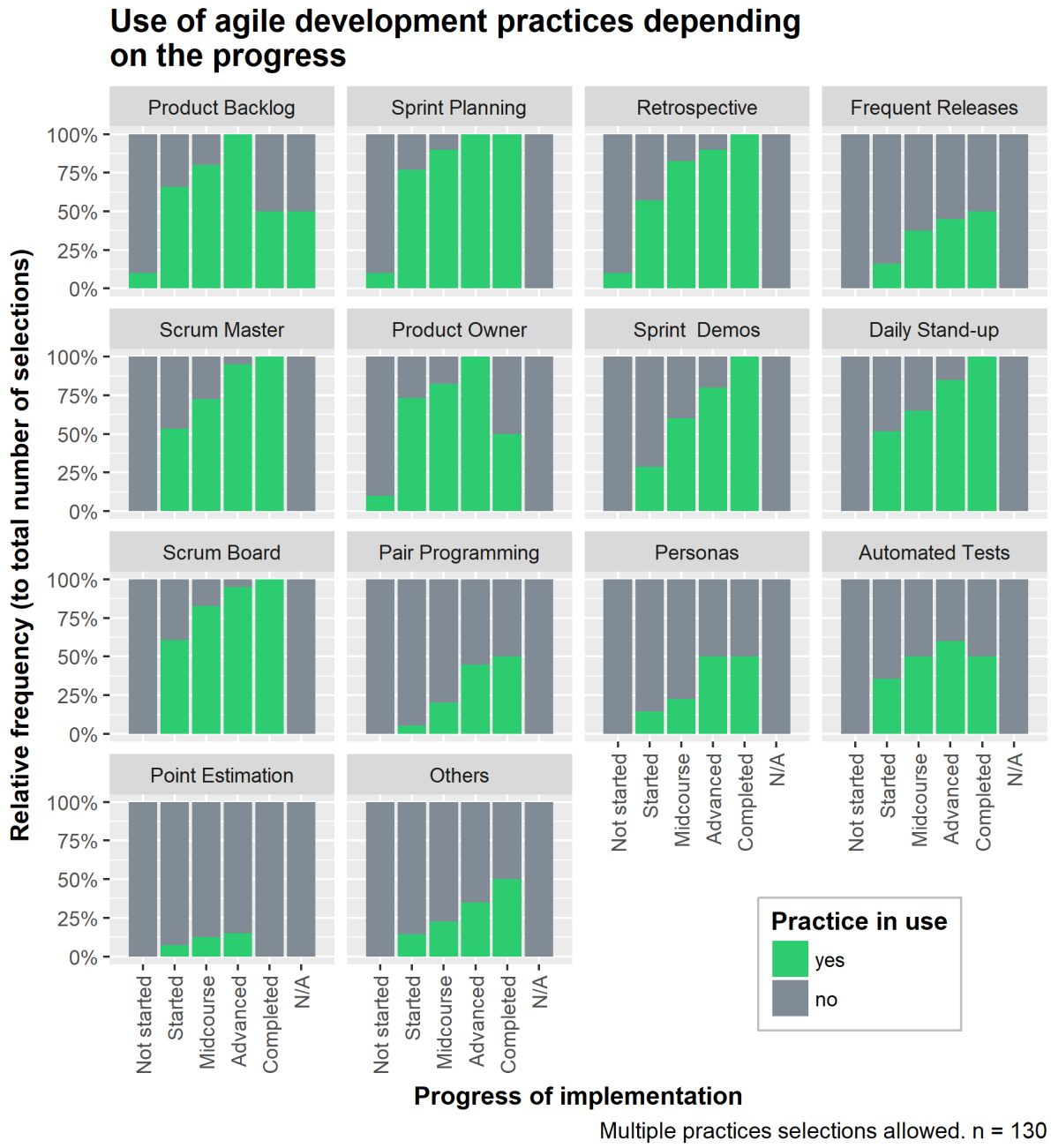


Figure 4.14.: Practices used in agile development of physical products in correlation with the implementation progress.

Description
Product backlog, sprint planning, retrospectives, Scrum master, product owner, daily stand-ups and Scrum boards are starters. More than 50% of the companies that just started implementing agile development have them already in use. The other practices gain importance rather late. However, no practice among those asked diminishes when companies get more advanced in agile development. Especially pair programming, personas, frequent releases and sprint demos, but also others are used primarily by those that are advanced already.

Among the participants only two said that they completely implemented agile development and another two chose N/A. The "completed" and "N/A" bar is negligible and are shown for the sake of completeness only. See Figure 7.9 for the number of participants per progress category.

Key learnings

- During implementation, more and more practices are in use. There is no practice, agile teams discard.
- Practices supporting operative project work (such as pair programming) gain importance rather late.

Interpretation

- The more companies are certain in applying agile methods, the more they use alternative practices (to Scrum).
- Potentially, practices designed for agile software development might be tricky to apply in agile hardware development. In particular, this might be true for design supporting practices like frequent releases or automated testing.

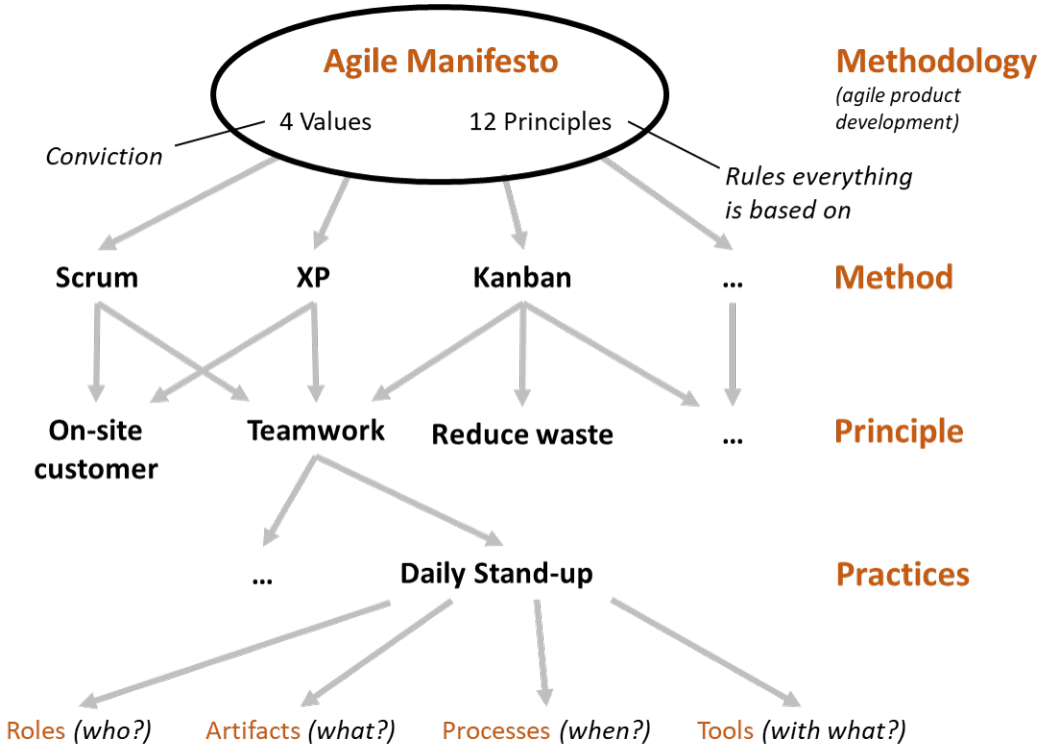


Figure 4.15.: Hypothetical framework to structure and define often used terms in the context of agile development.

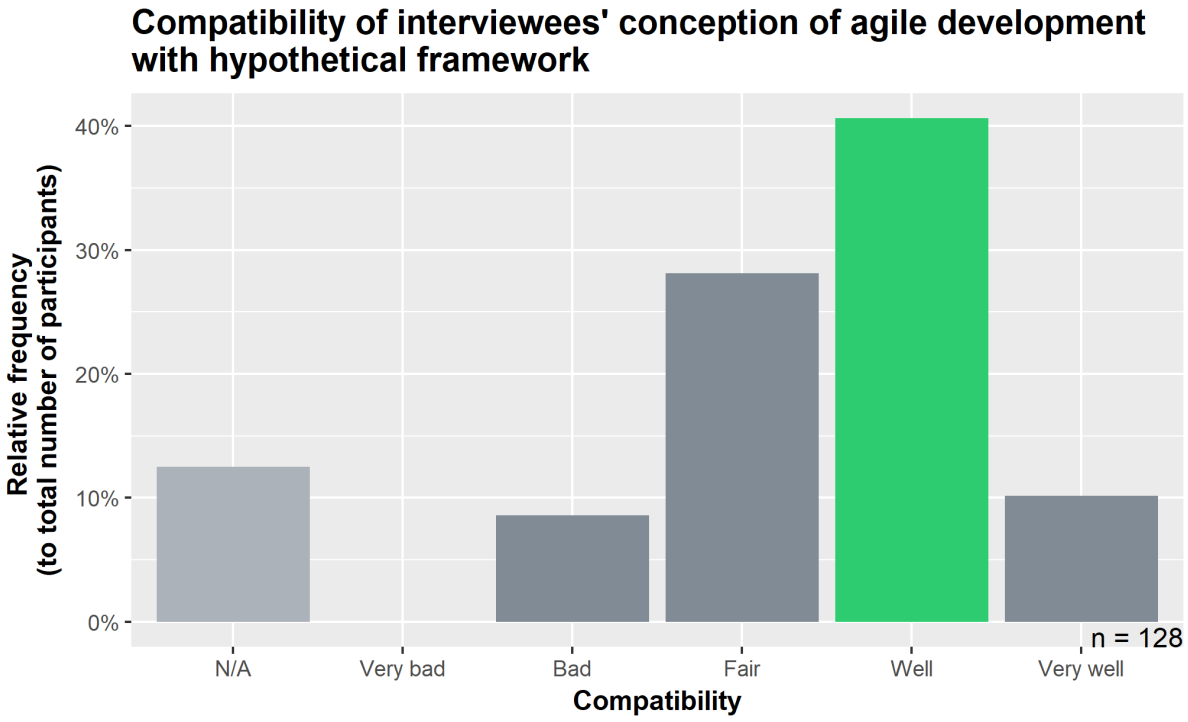


Figure 4.16.: Compatibility of interviewees' conception of agile development with above hypothetical framework.

Description
Terms like method, principles and practices are often used, but the same term seems to have different meanings depending on who uses it. In order to support defining and integrating popular terms in the context of agile development, the authors of the study at hand set up a hypothetical framework to test its compatibility with the conception of practitioners.

It is assumed that agile development predominantly is a mindset, which is specified in the Manifesto of Agile Software Development (Beck et al. 2001). There, four values (e.g. "Individuals and interactions over processes and tools") and twelve principles (e.g. "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.") form the very basics of agile development. It is assumed that acting according to the Manifesto in development means to work in an agile manner. This is the methodology of agile development.

In general, a methodology consists of several methods. Applied to agile development, Scrum, Kanban, Design Thinking etc. are methods that contain partially overlapping elements. Those elements are guiding principles that constitute the method's character. Among others, Scrum builds on on-site customers and teamwork. Principles again can be detailed in practices that are applicable on a daily basis. Practices like daily stand-up, for instance, as one way to put the teamwork principle into effect implies roles (Who acts?), artifacts (What needs to be done?), processes (When and how can it be done?) and tools (By what means?).

In German, the study uses "Grundsätze" for the principles of the Manifesto and "Prinzip" for guiding principle of the methods. The graphic has been shown to the participants (in German) to obtain their assessment to which degree they think that the hypothetical framework is compatible with their conception of agile development. The participants were asked to assess according to their experience.

Overall, the survey reveals that the hypothetical framework is compatible to the conception in practice to a large extent as the frequency plot is left-skewed and the mean equals 3.6 out of 5 rating options (excl. N/A). Eleven participants left comments in a free text field. While, as expected, some report that they use other terms (e.g. framework or procedure model instead of method; didactics or philosophy instead of methodology), others respond that agile development is a mindset which the framework does not emphasize enough. However, the fundamental structure has not been criticized in the free text field, participants rather confirm that they have established a comparable structure.

Key learnings
- The Manifesto is understood as the overarching working philosophy, which agile methods as guiding frameworks build on and agile practices concretize practically.
- An appropriate mindset seems to be the core of agile development. Methods like Scrum, principles like teamwork and practices like daily stand-ups build upon it, but leave room for many other ways to act according to the mindset.

Interpretation
- It seems likely that practitioners are aware of the necessity to interpret and select agile methods and practices according to the specific development context.
- It remains untested whether survey participants understood presented framework well enough before evaluating it.
- It also remains open at which point in the presented framework ambiguous interpretations can evolve. The survey tests the overall agreement only.

Interim Conclusion

The Manifesto of agile development is considered as overarching philosophy for organizing a development project in practice in a very lively and flexible way (compare also Figure 4.1). Based on that, agile methods that contain agile practices are defined that, in turn, give practitioners guidance to live up to the required mindset.

Scrum, Kanban and Design Thinking are the most popular and most visible methods in agile hardware development. Those methods and their corresponding practices serve as starting points for most beginners in becoming agile. The more experienced a company becomes in agile hardware development, the more they differentiate agile methods and practice in detail. They become picky about which roles, artifacts, processes and tools really brings value for the development team. It is likely that experienced companies have selected and adapted agile practices according to their specific context and development tasks.

Moreover, project management-oriented methods and practices (e.g. Scrum and Kanban) are used earlier than design-oriented methods like Feature or Test Driven Development as well as Design Thinking. Companies seem to start in improving administrative project activities (organizing, coordinating, steering) when coping dynamic development conditions. However, agile development is perceived as more beneficial for operative project work (designing), as visible in Figure 4.3.

5. Motivations to Implement Agile Development

This chapter deals with reasons why companies want to become agile in their development of physical products. They associate certain benefits with the concept of agile development and expect to reach them by applying guiding frameworks like Scrum. In order to investigate the expectations, that source companies' motivations to implement agile hardware development, the author team collected 23 potential partial values. One the one hand, this value collection bases on the practical experiences with agile development of the author team. On the other hand, the list of partial values contains often referred to benefits that are used, for instance, to make sell the concept (e.g. "Agile hardware development makes the project cheaper").

For that, the participants were asked to evaluate their perception on a 5-point Likert scale. 0 refers to "hardly any value" and 4 means "large value". The survey question was: Which value did or do you expect from agile hardware development?

As experienced participants might be biased by the real effects of agile hardware development, diagrams in this study exclude companies claiming a "midcourse", "advanced" or "completed" implementation progress. Those companies made the adoption decision years ago. Including them might bring the expected value closer to the real value of agile development because their ratings might include learning effects. Thus, ratings for expected improvements of agile hardware development in this study are sourced by beginners only.

Firstly, the study sheds light on companies' expectations and motivations why to implement agile development of physical products. Secondly, comparisons with existing empirical studies are drawn. The results of the study at hand are compared to agile software development by means of Version One (2016), and to agile organizations in general by means of Komus et al. (2017).

5.1. Expected Benefits Through Agile Development

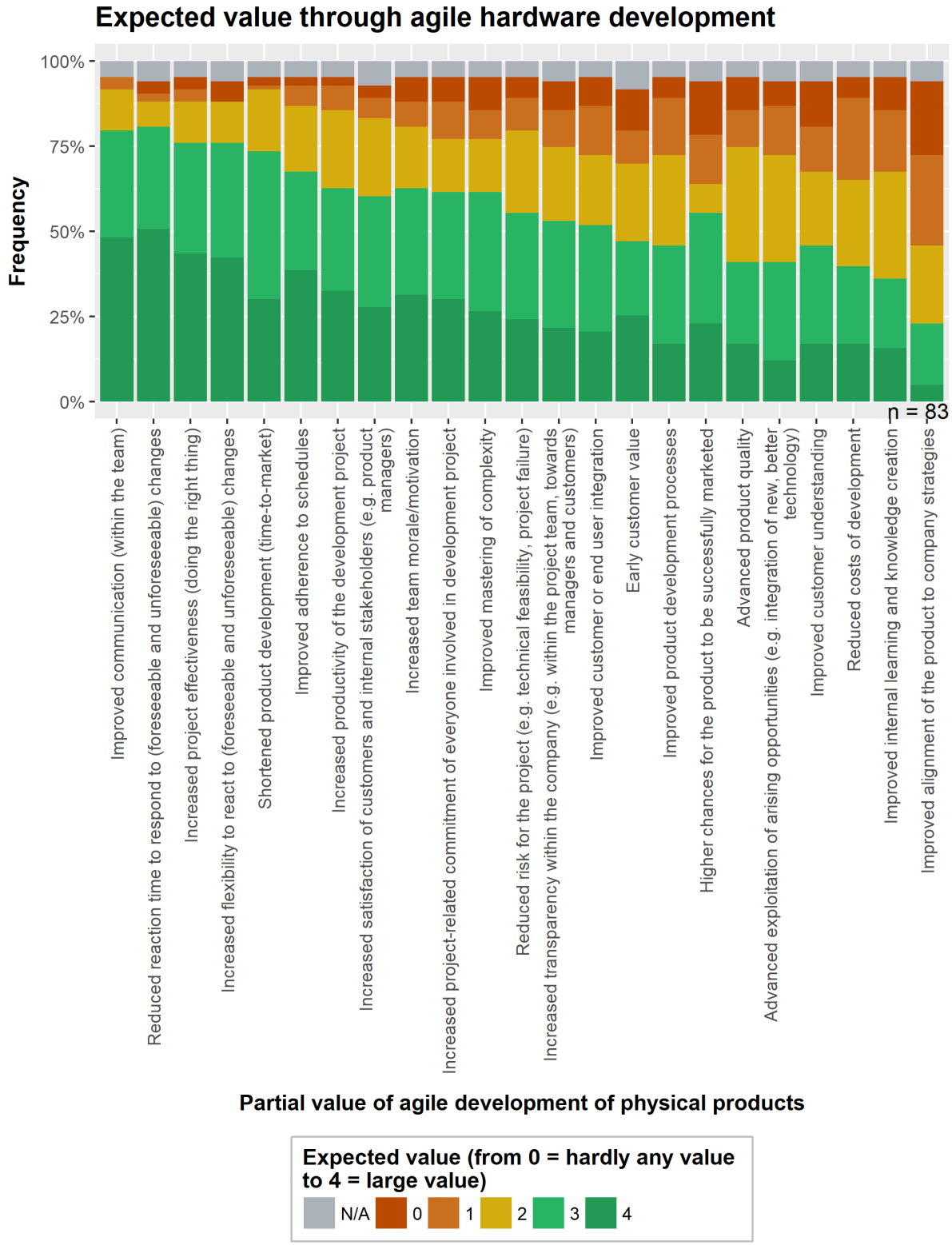


Figure 5.1.: Expected value of agile development of physical products sorted by decreasing means.

Description
In this graph, statements about the values of agile hardware development are sorted by how large the targeted value is. On the left, there are the statements which are deemed to have the most desired value and on the right there are the statements from which most companies do not expect much benefit. In general, only very few see no value in any of these statements and most participants value them somewhere between 4 and 2.

Only beginners are displayed (not yet started, started and N/A's). Answers of experienced companies include learning effects already. The chance that companies that are engaged in agile development for years already unconsciously align their reasons for adopting to the real improvements is high. This bias is considered by excluding experienced companies.

Key learnings
- Desired improvements through agile development are manifold.
- More than 75% expect agile development to improve communication, reaction time to changes, flexibility and project effectiveness.
- Improved adherence to schedules, shortened time-to-market and increased productivity are among the Top 7. Expectations on agile development to have positive effects on classical KPI's are relatively high.

Interpretation
- Most participants have high and widespread expectations.
- List of partial values can be not exhaustive.

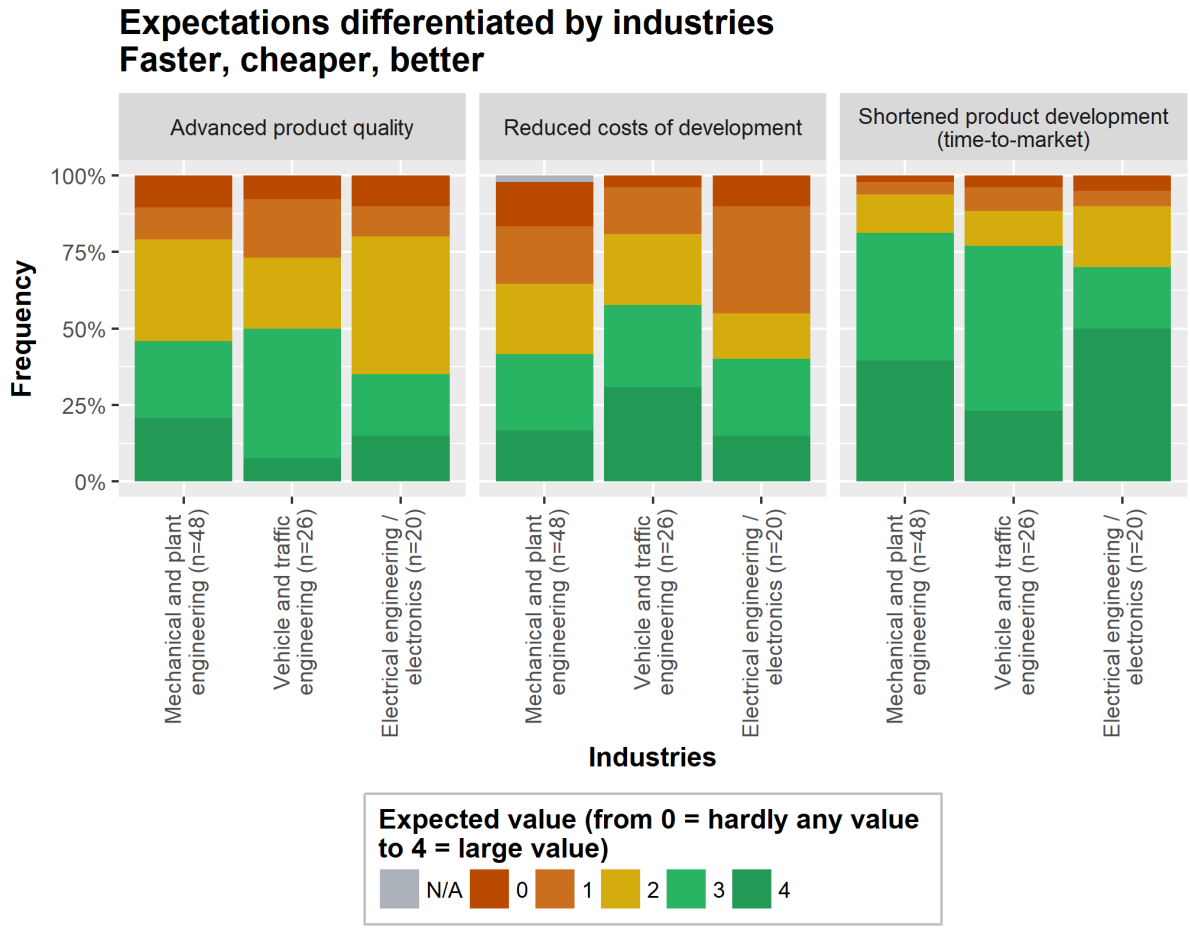


Figure 5.2.: Desired value for classical controlling KPI's compared among industries.

Description
A controlling KPI selection and their variation between industries is shown. As the total number of participants are the highest in mechanical and plant engineering (ME), vehicle and traffic engineering (VE) and electrical engineering (EE), they are used for the sake of comparison.

While low rating frequencies do not vary a lot for product quality and time-to-market, there are larger differences between the medium and the two highest ratings 3 and 4. On average, VE expects slightly more improvements in product quality and development costs than ME and EE. Among these controlling KPI's, the desire for benefits in development lead times is highest.

Key learnings
- 30 - 50% expect agile development to improve quality and costs, only 20 - 30% do not.
- About 70% think that agile development accelerates the projects, less than 12% do not.
- Smaller differences among industries exist concerning classical controlling KPI's.
- ME seems to be unsure about the value concerning product quality and development costs as the frequencies are quite evenly distributed.

Interpretation
- Differences in industry-dependent value expectations might stem from different context circumstances such as cost pressure and demand for innovation. Companies might read some improvements into agile development that would be worthy for their circumstances, but perhaps not achievable with the particular approach.

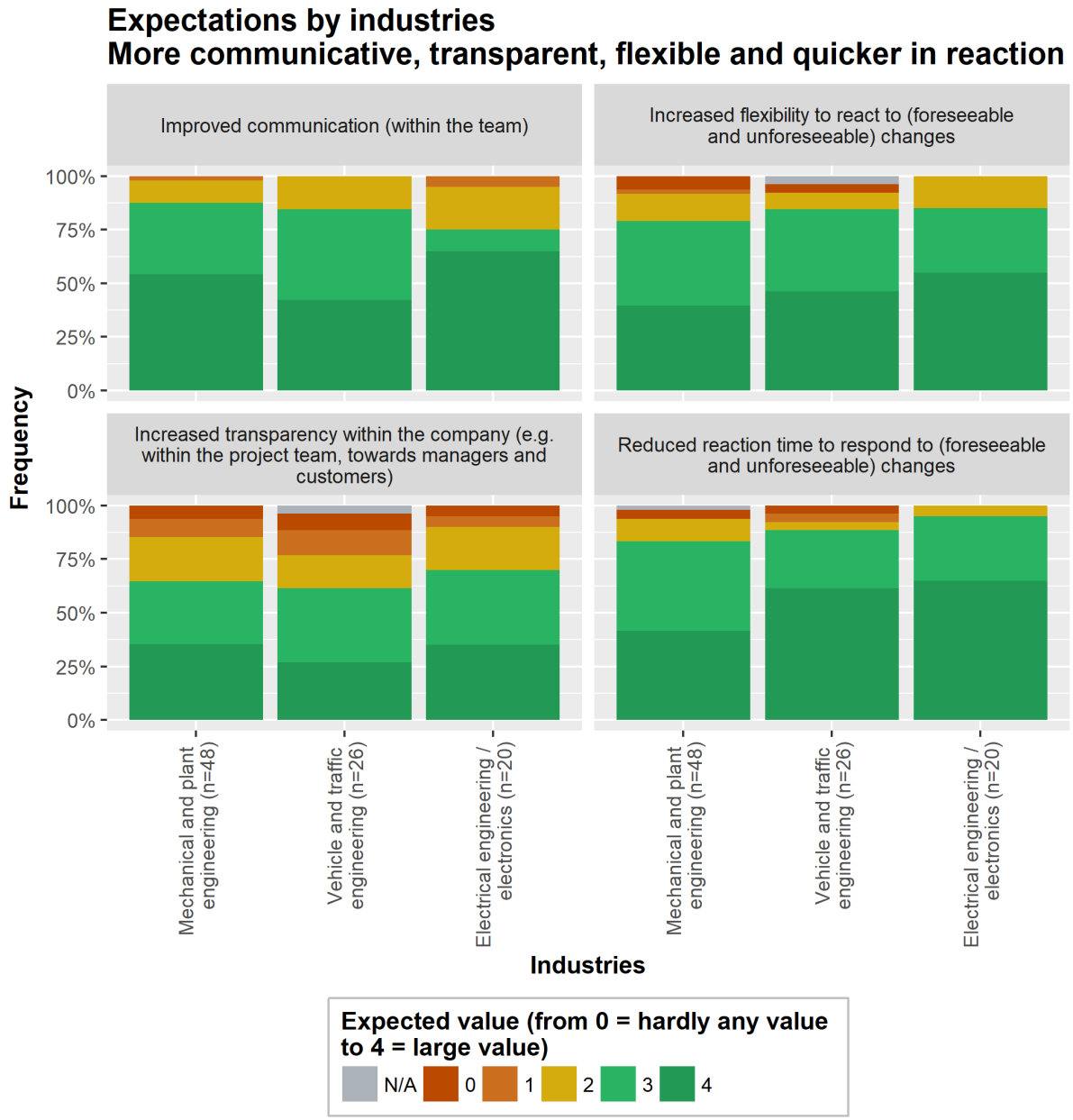


Figure 5.3.: Desired value for soft aspects compared among industries.

Description
Similarly to the previous graph it depicts a selection of expected values through agile hardware development. It limits itself on the Top 4 real improvements (compare Figure 6.1).

Obviously, communication, reaction and flexibility is targeted as great benefits caused by agile development for all three industries. Almost no one has concerns about less improvements here. Ratings for increased transparency are less optimistic. Much more (about 70%) of all participants from EE highly desire improved communication - much more than the other two industries. Analogously, ME targets at less improvements in reaction time to changes than the other two industries.

Experienced companies might be biased because they have experienced the real effects already. The chance that companies that are engaged in agile development for years already unconsciously

align their reasons for adopting to the real improvements is high. As the total number of participants per industry would diminish, which causes statistical insignificance, this diagram considers all participants.

Key learnings

- Differences among industries exist for soft aspects, but they are even smaller than for classical controlling KPI's.
- All industries presented have higher expectations for shortened time-to-market than for transparency.
- More than 75% want to implement agile hardware development in order to improve communication, increase flexibility and reduce reaction time to changes.

Interpretation

- Transparency seems to be perceived by practitioners as an adequate means to cope with complexity in product development projects.
- Practitioners that think about becoming agile in product development are in need of high flexibility and a fast reaction rate to changes. They might be confronted with a very dynamic development environment which makes this ability necessary.

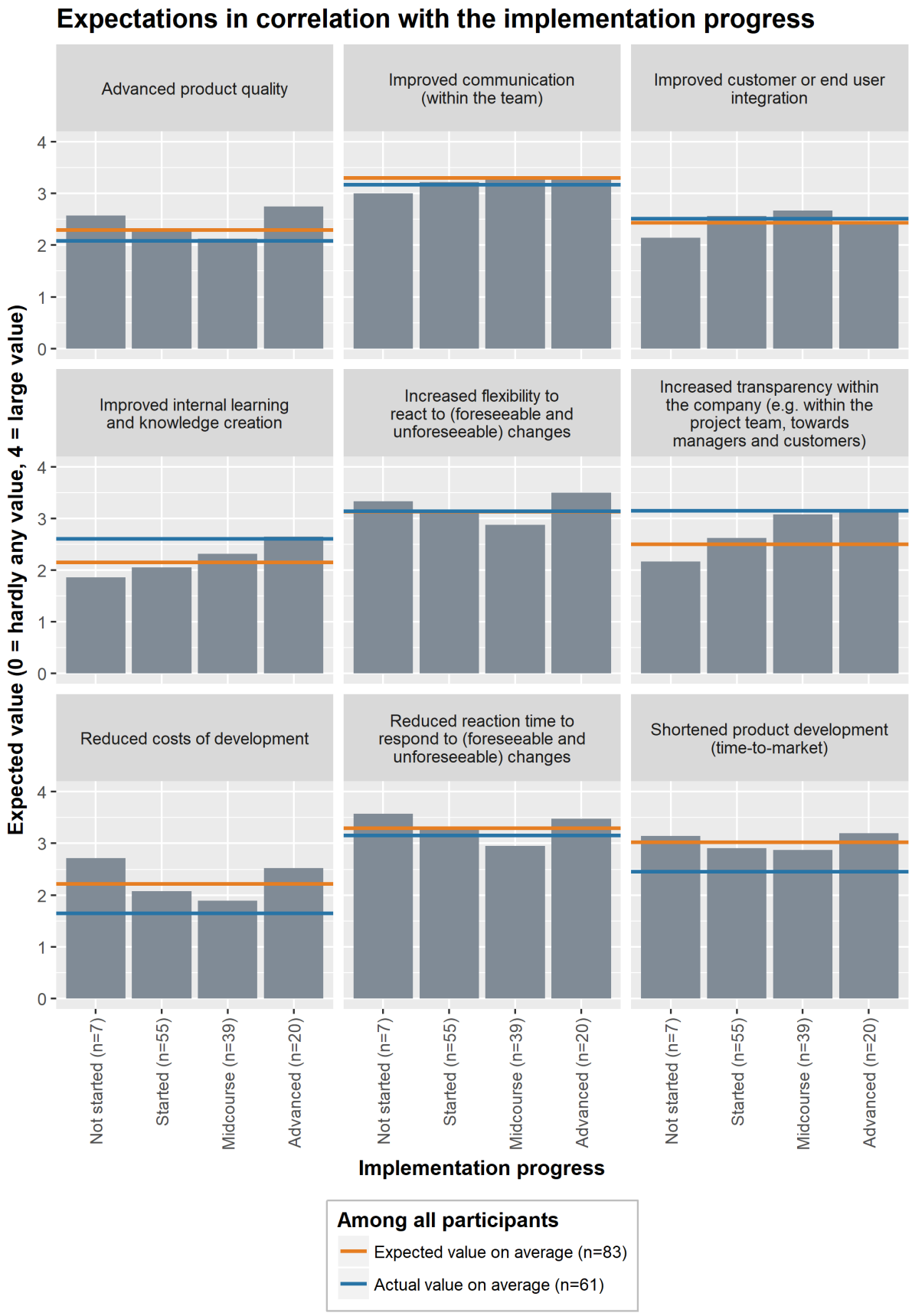


Figure 5.4.: Implementation progress in correlation with the expected value of agile hardware development.

Description
Figure 5.4 shows classical controlling KPI's, the Top 4 real improvements (compare Figure 6.1) and two more selected values in correlation with the companies' implementation progress. It investigates the learning effects. For that, the diagram displays the mean per implementation progress as bars and the overall means among all participants as horizontal lines according to Figure 6.1. It also represents the real value of agile development that will be explained in Chapter 6 "Potentials of Agile Development".

The more advanced a company is in developing physical products in an agile way, the less they are deemed to be motivated by product quality, development costs or project lead times. However, the advanced progress category does not follow this trend for neither value. Similar effects are visible for flexibility and reaction time. In contrast to that, companies that have already been engaged in agile development for years were more motivated by improving knowledge creation, team communication and transparency than those companies that have just started.

Experienced companies might be biased because they have experienced the real effects already. The chance that companies that have already been engaged in agile development for years unconsciously align their reasons for adopting to the real improvements is high. As the total number of participants per outsourcing degree would diminish, which causes statistical insignificance, this diagram considers all participants.

Key learnings
- While experienced companies are closest to the real improvements, beginners are the farthest away.
- Experienced companies are less driven by classical KPI's.
- Companies that stated that they have an advanced implementation progress do not follow the general upward or downward trend formed by the other three categories. It leads to saddle-shape trend lines for some aspects (such as quality and costs).

Interpretation
- Learning curve effects are present in the diagram. Companies being experienced in agile hardware development have learned what the approach really improves and aligned their current motivation to still increase agility accordingly. Similarly or alternatively, companies that start implementing agile development of physical product development today have different motivations to become more agile than their forerunners.
- Hype effect, which is present in Figure 6.2, becomes even more considerable when apportioning the answers to the implementation progress.

Interim Conclusion

Practitioners have high expectations on agile hardware development. They want to become more agile because they want to improve communication, reduce reaction time to changes, increase project effectiveness, shorten product development and improve the adherence to schedule. Thereby, classical KPI's are deemed relatively high. In contrast, improved alignment of the product to company strategies, improved internal learning and knowledge creation, reduced costs of development, improved customer understanding and advanced exploitation of arising opportunities are minor motivations to implement agile hardware development.

While the data reveals less variance due to industry affiliation, there are major differences due to the implementation progress. In the majority of cases, experienced companies are closer to the real value of agile hardware development, beginners the farthest. Companies that just started to implement agile hardware development are less motivated, for instance, by increased transparency, improved internal learning and knowledge creation and improved communication than those companies that are already experienced in agile hardware development.

5.2. Comparisons with Version One (2016) and Komus et al. (2017)

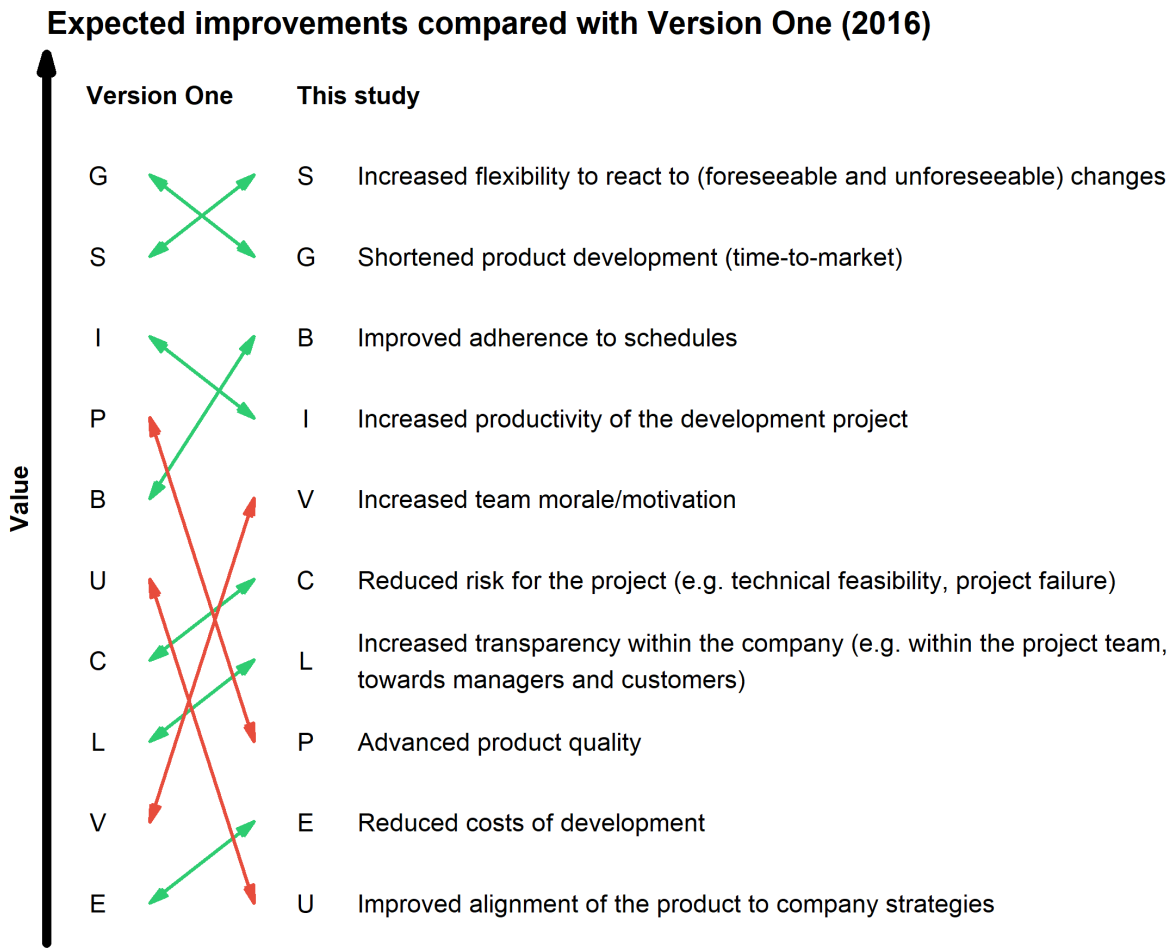


Figure 5.5.: Comparison of results from Version One (2016) and the study at hand.

Description
The diagram contrasts the results on expected improvements derived by the well-known study of Version One (2016) and by the study at hand. While Version One focuses on agile software development, the study at hand investigates agile hardware development. As the list of partial values tested by Version One and the study at hand does not overlap completely, the diagram shows only those that were evaluated in both studies. The letter ID's used are randomly sorted and are consistent with other diagrams in this document (compare Table 6.1).

It is important to note that the survey of Version One asked the interviewees to select the partial values that have influenced their decision to implement agile software development most (yes/no). In contrast, the study at hand asked the participants to rate each single partial value, to which degree they think it will reveal as improvement (5-point Likert scale). However, the derived rankings of each study are comparable. This is what the diagram depicts. Minor differences in the rankings are marked green. Major differences ($>$ 3 ranks) are colored red.

Key learnings
- In general, companies both from software and hardware industries are motivated to implement agile development with the expected improvements being quite similar.

- However, companies from the software industry expect agile software development to have a higher impact on product quality and alignment of the product to company strategies.
- Companies from the hardware industry think that agile hardware development has a higher positive influence on the team morale than it is perceived in agile software development.

Interpretation

- Although some differences between the Version One study and the study at hand exist, quite similar results were derived by both studies in general.
- Differences could stem from advanced learning curve effects as agile software development has become a more mature concept already.

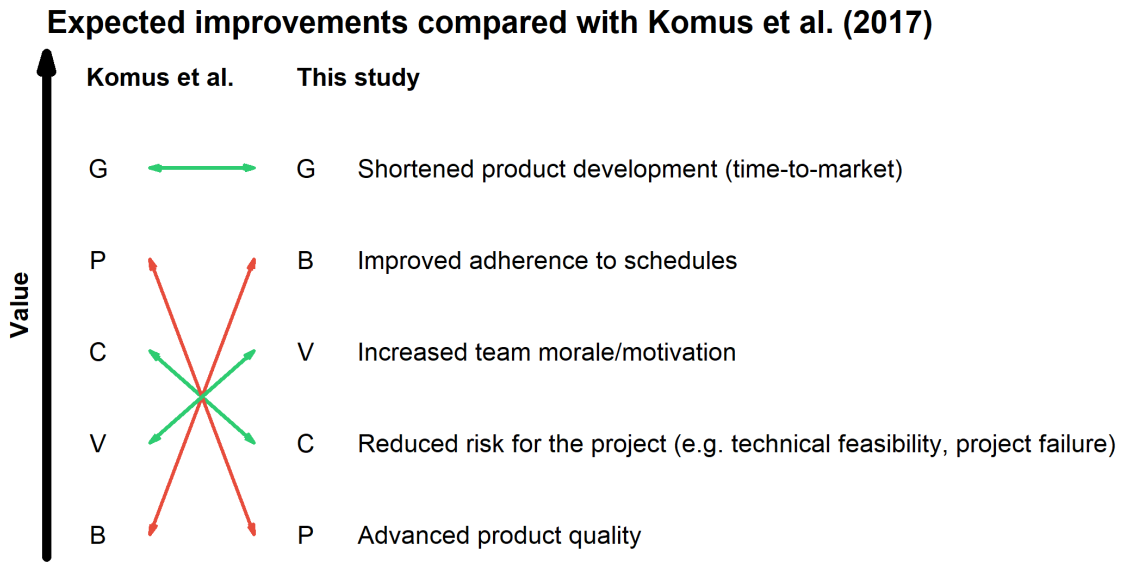


Figure 5.6.: Comparison of results from Komus et al. (2017) and the study at hand.

Description
The diagram contrasts the results on expected improvements derived by the German large scale study of Komus et al. (2017) and by the study at hand. While Komus et al. focus on agile organizations and agile project management, the study at hand investigates agile hardware development. As the list of partial values tested by Komus et al. and the study at hand does not overlap completely, the diagram shows only those that were evaluated in both studies. The letter ID's used are randomly sorted and are consistent with other diagrams in this document (compare Table 6.1).

It is important to note that the survey of Komus et al. asked the interviewees to select the partial values that have influenced their decision to implement agile software development most (yes/no). In contrast, the study at hand asked the participants to rate each single partial value, to which degree they think it will reveal as improvement (5-point Likert scale). However, the derived rankings of each study are comparable. This is what the diagram depicts. Minor differences in the rankings are marked green. Major differences ($>$ 3 ranks) are colored red.

Key learnings
- Companies implementing agile organizations or agile project management are less motivated by improving adherence to schedules, but more in advancing product quality.

Interpretation
- Although some difference between the Komus et al. study and the study at hand exist, quite similar results were derived by both studies in general.
- Important partial values that are among the Top 3 of expected improvements through agile development (such as increase flexibility) have not explicitly been tested by the study of Komus et al..

Interim Conclusion

Although the well-known empirical studies of Version One (2016) and Komus et al. (2017) do not test the same list of partial values, some overlaps exist. Previous diagrams contrasted the results of these studies with the results of the study at hand. It turns out that differences are present in some aspects, but they are surprisingly similar in general.

Differences in the results concerning the desired value of agility might be caused by the different fields of application. While Version One (2016) focuses on agile software development and Komus et al. (2017) investigates agile organizations and agile project management, the study at hand limits itself to agile development of physical products.

6. Potentials of Agile Development

After knowing what practitioners drives to implement agile development of physical products in their companies, it still remains open whether agile development can live up to its perceived promise. In contrast to the previous chapter, the following diagrams focus on real effects only. For that, interviewees were asked to evaluate the same collection of 23 potential partial values with the same 5-point Likert scale as used in Chapter 5 "Motivations to Implement Agile Development". Now, the participants rated based on their experience with agile hardware development - in retrospect: The survey question was: Which actual improvements have you really achieved through agile hardware development?

This facilitates direct comparisons between expected and real benefits of agile hardware development. In turn, this comparison implies details for the hype effect of agile development and can explain in which aspects agile development is overestimated/inflated or underestimated.

As the author team assumes that participants from companies that have just started implementing agile development cannot evaluate actual effects reliably, they are excluded from the data subset for all real value graphs in this study. As a logical consequence, the total group of participants is divided into two groups: Beginners are those that ticked "not started" or "started", and experienced companies are those that claim to be "midcourse", "advanced" or "completed" in implementing agile hardware development. Thus, the groups are mutual exclusive and the sum of both groups equals the total group of survey participants (minus those that are excluded due to other reasons such as incomplete answers).

Firstly, the study sheds light on the real improvements through agile development of physical products and contrasts this insights with companies' expectations and motivations why to implement agile development. Secondly, comparisons with an existing empirical study are drawn. The results of the study at hand are compared to agile software development by means of Version One (2016).

6.1. Real Effects versus the Hype of Agile Development

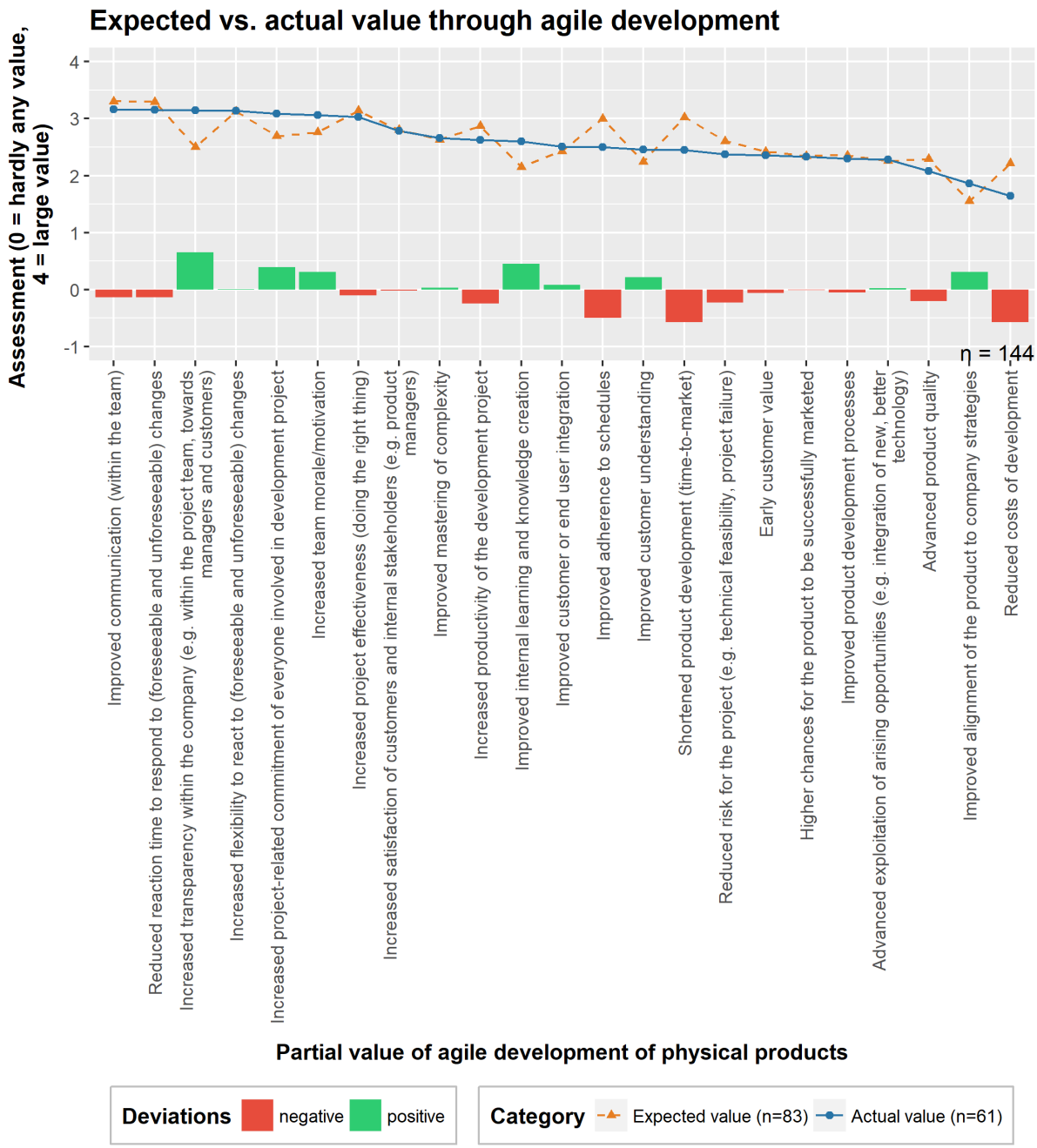


Figure 6.1.: Expected vs. actual value through agile hardware development sorted by the real value.

Description

The diagram depicts the means of each partial value investigated and outlines the delta between expected and real effects. All survey participants were asked to rate their expectations. However, a bias due to experience exists: Experienced companies might align their original adoption motivations according to their experience they did not have when they made the implementation decision. That is why the expected value graph bases on responses given by beginners and the real value graph by experienced companies only. Consequently, the expectation graph is the same as the data presented in Figure 5.1, the real value graph as well as the delta between the

two graphs are new here. Tested categories are sorted by the real value.

Key learnings

- The real improvements of agile development of physical products is manifold.
- Real improvements exists especially in soft aspects, namely improved communication, reduced reaction time to changes, increased transparency, increased flexibility, improved project-related commitment, increased team morale and project effectiveness. These aspects reach means greater than 3 (with 4 as maximum).
- Although agile hardware development can cause improvements in development costs, alignment of the product to companies strategies and product quality, they are rather negligible compared to the other values.

Interpretation

- Especially reduced costs, shortened time-to-market and improved adherence to schedules seems to be irritating for companies. Many have high expectations of agile development in these aspects, but they appear in the lower half of the real value ranking.

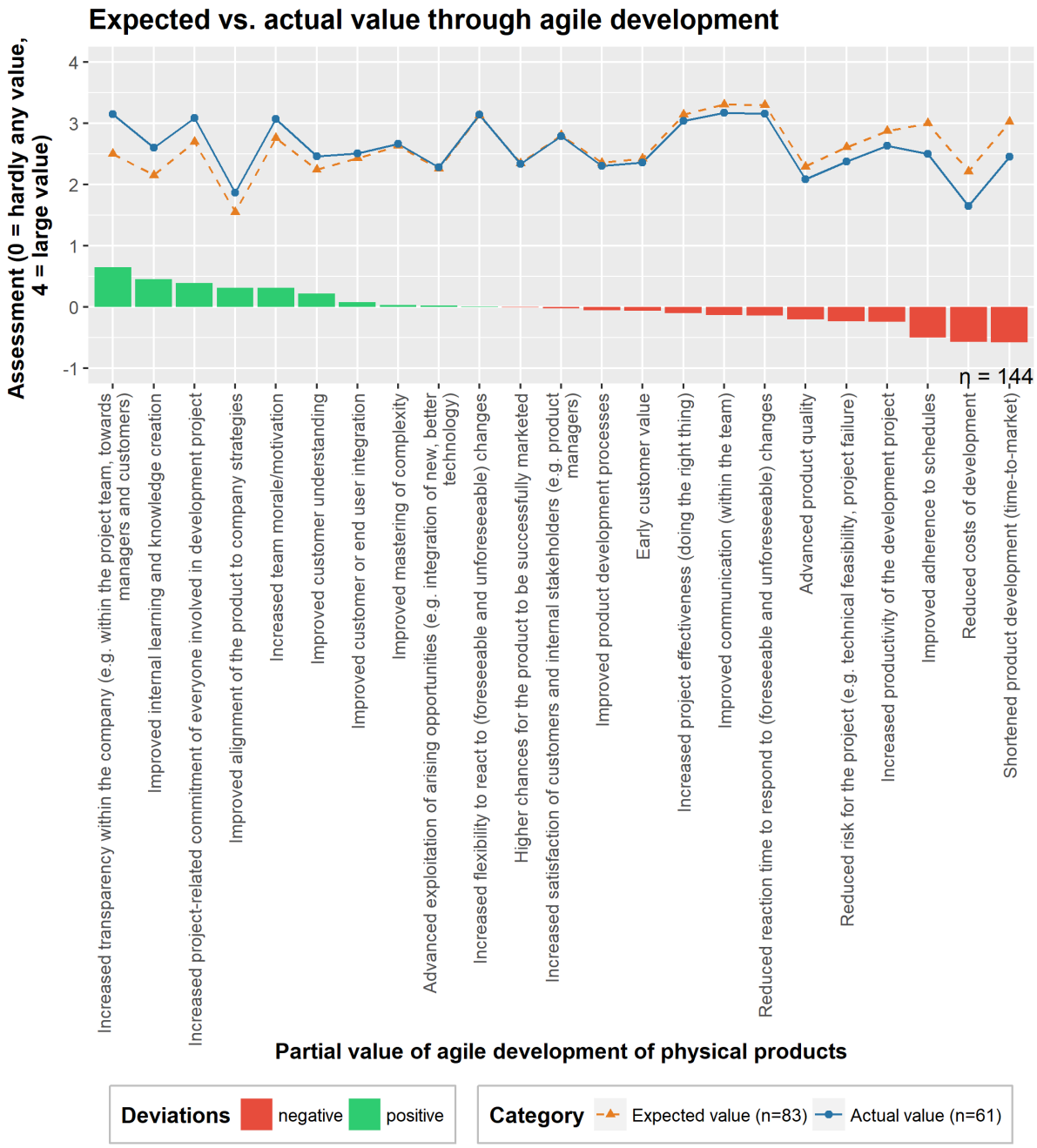


Figure 6.2.: Expected vs. actual value through agile hardware development sorted by the delta.

Description

The diagram depicts the means of each partial value investigated and outlines the delta between expected and real effects. All survey participants were asked to rate their expectations. However, a bias due to experience exists: Experienced companies might unconsciously align their original adoption motivations according to their experience they did not have when they made the implementation decision. That is why the expected value graph bases on responses given by beginners and the real value graph by experienced companies only. In contrast to the previous diagram, tested categories are sorted by the delta to separate underestimated and overestimated partial values of agile hardware development.

Key learnings

- Expectations fit very well to real values of agile hardware development when it comes to im-

proved mastering of complexity, advanced exploitation of arising opportunities, increased flexibility, higher chances to be successfully marketed as well as increased satisfaction of customers and internal stakeholders.
- Among the partial values tested, there are more that are overestimated than underestimated.
- Especially soft aspects, that are usually not quantitatively measurable, such as transparency and knowledge creation, are underestimated.
- Especially hard aspects, that are usually well quantifiable, such as project lead times, costs of development and adherence to schedules, are overestimated.

Interpretation
- The hype about agile development seems to exist particularly in hard controlling KPI's. A hype causes frustration when one realizes that the desired value is not achievable in that way.
- There is a serious risk that wrong expectations due to the hype leave scorched earth. Although agile development can cause considerable improvements, the approach could be abandoned because some companies used it with the wrong purpose.
- Aspects that have statistically negligible deltas (such as increased flexibility or higher chance for the product to be successfully marketed) are not hyped (neither in a positive nor negative way).

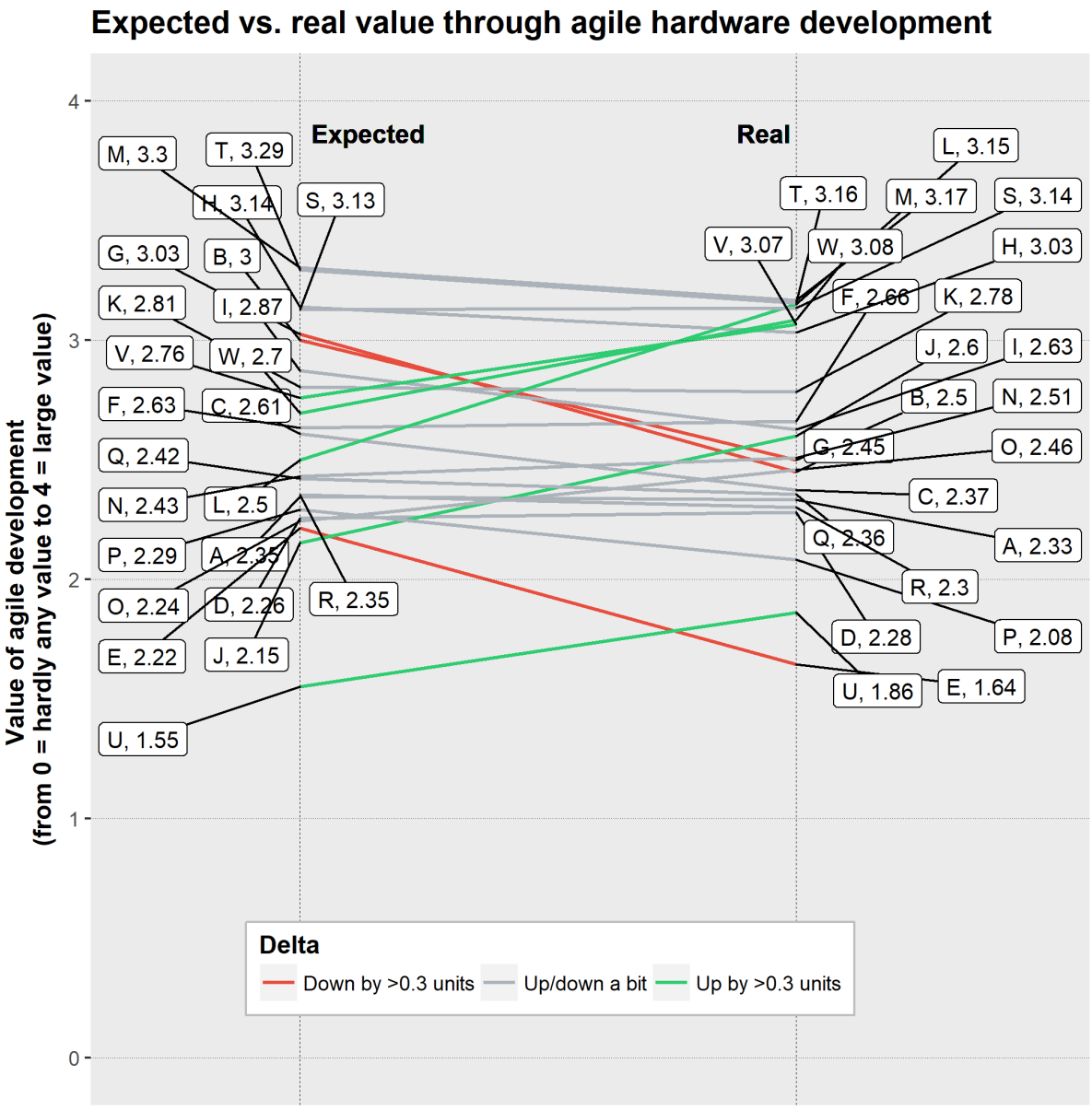


Figure 6.3.: Changes in value ranking - expected vs. real value through agile hardware development. Compare ID's Table 6.1 on next page.

Description
The diagram compares the means and ranking of all partial values of agile hardware development investigated on both the expected and the real improvement dimension. It contains the same data as Figure 6.1 and Figure 6.2, but visualizes it differently. Please compare the ID's with Table 6.1. Differences in more than 0.3 units are colored red and green depending on their direction.

While increased transparency as well as improved internal learning and knowledge creation are ranked much higher among the real values, increased productivity, improved adherence to schedules, shortened time-to-market and reduced development costs fall back drastically.

Key learnings

- All partial values investigated indeed provide improvements on average. Except of development costs, they are all higher than 2 symbolizing the middle of the scale.
- Especially classical controlling KPI's are a lot overestimated, while soft factors like transparency and knowledge creation are a lot underestimated. Companies profit more of the latter than by the former when adopting agile development.

Interpretation

- The diagram manifests the interpretations from Figure 6.1 and Figure 6.2.

Table 6.1.: Values investigated and their ID's used in the study at hand. Randomly sorted.

ID	Value investigated
A	Higher chances for the product to be successfully marketed.
B	Improved adherence to schedules.
C	Reduced risk for the project (e.g. technical feasibility, project failure).
D	Advanced exploitation of arising opportunities (e.g. integration of new, better technology).
E	Reduced costs of development.
F	Improved mastering of complexity.
G	Shortened product development (time-to-market).
H	Increased project effectiveness (doing the right thing).
I	Increased productivity of the development project.
J	Improved internal learning and knowledge creation.
K	Increased satisfaction of customers and internal stakeholders (e.g. product managers).
L	Increased transparency within the company (e.g. within the project team, towards managers and customers).
M	Improved communication (within the team).
N	Improved customer or end user integration.
O	Improved customer understanding.
P	Advanced product quality.
Q	Early customer value.
R	Improved product development processes.
S	Increased flexibility to react to (foreseeable and unforeseeable) changes.
T	Reduced reaction time to respond to (foreseeable and unforeseeable) changes.
U	Improved alignment of the product to company strategies.
V	Increased team morale/motivation.
W	Increased project-related commitment of everyone involved in development project.

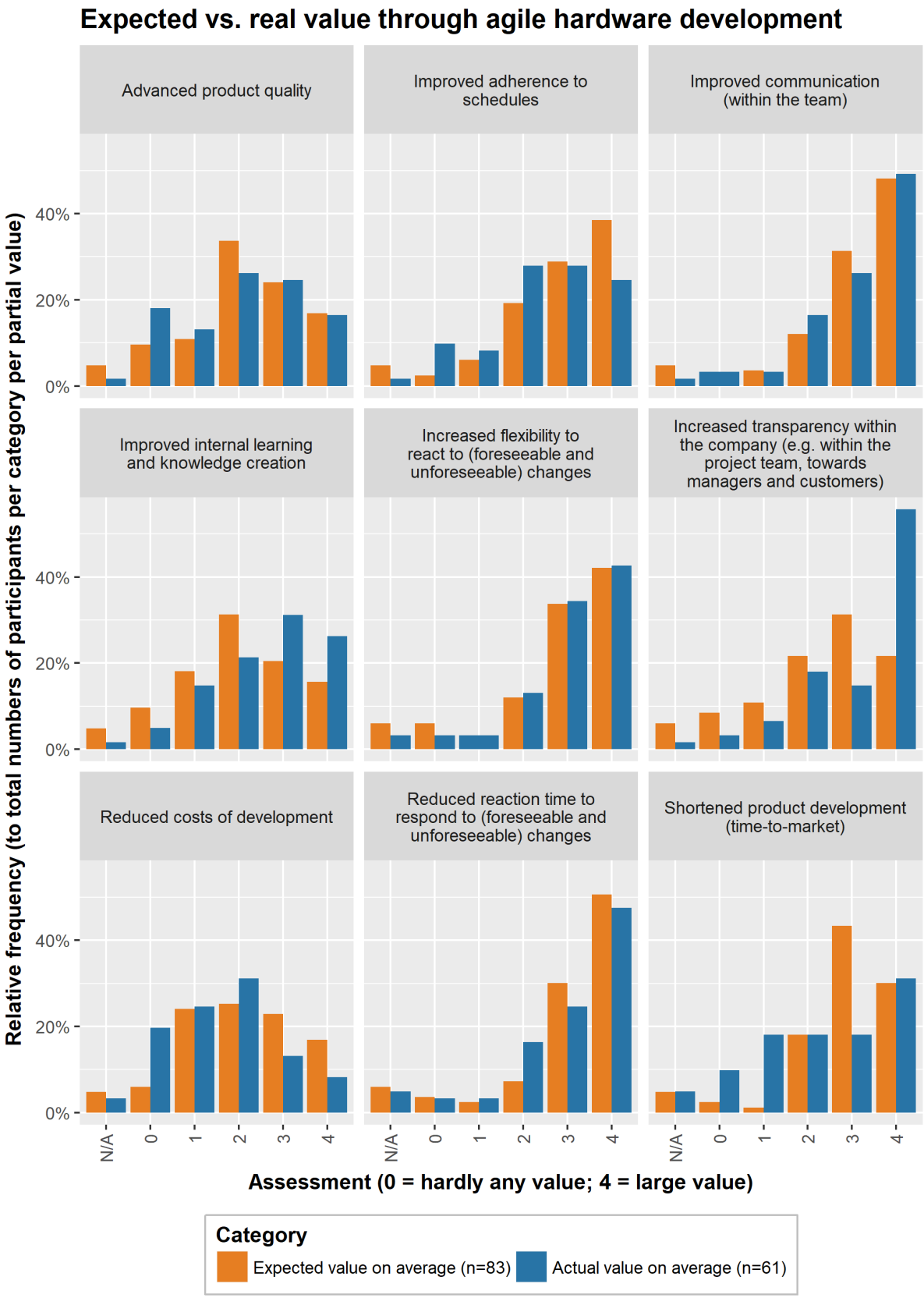


Figure 6.4.: Uncertainty and differences in selected partial values of agile hardware development.

Description
Differences between expected and actual improvements through agile hardware development are displayed in selected partial values. While the participants are very certain about the effects on communication, flexibility, reaction time to changes and transparency (as the distributions have a small standard deviation within the single graph), they are less certain about product quality, costs, time-to-market and internal learning. However, internal learning graphs are similarly distributed, but the actual value graph is shifted to the right by about 1 unit. Participants desire improved adherence to schedules and shortened project lead times, but when it comes to the actual improvement, the effects of agile development in these aspects are relativized.

Key learnings
- In most cases, the ratings about actual effects are more certain as standard deviations are smaller.
- Especially increased learning and knowledge creation, shortened product development as well as increased transparency shift largely.

Interpretation
- Improvements in partial values like product quality, development costs, time-to-market and adherence to schedules do not seem to be obvious. Even experienced companies in agile development, that rated for the real values only, are uncertain about it. There might be improvements in some cases, but by far not in all.
- High desires in reduced reaction time and flexibility show remarkable hardship in coping with the unforeseen.

Interim Conclusion

As shown in Chapter 5 "Motivations to Implement Agile Development", hard management expectations, which reflect well measurable KPI's such as costs, time and quality, are the focus of the implementation and application of agile development. However, agile development brings benefits in flexibility, reaction time to changes, transparency and communication.

Misleading expectations light the hype effect and causes frustrations that become present during the implementation. Thus, there is a risk that agile development will be classified as not conducive and, in turn, gets abandoned by practitioners. Yet, considerable potential would get lost for companies as the benefit of agile hardware development for soft factors is remarkable. In another study, the author team found out that soft factors influence hard KPI's on several means-to-an-end instances (Schmidt, Weiss, and Paetzold 2018).

It is important to be aware of the fact that the development task as well as the context conditions are highly decisive whether agile hardware development is worthwhile or not. Agile development is not a silver bullet and there are certainly contexts in which classical waterfall-like development is more efficient. But agile development can be an appropriate approach for uncertain and ever-changing development contexts in which, for instance, the project goal is not clearly describable.

6.2. Comparison with Version One (2016)

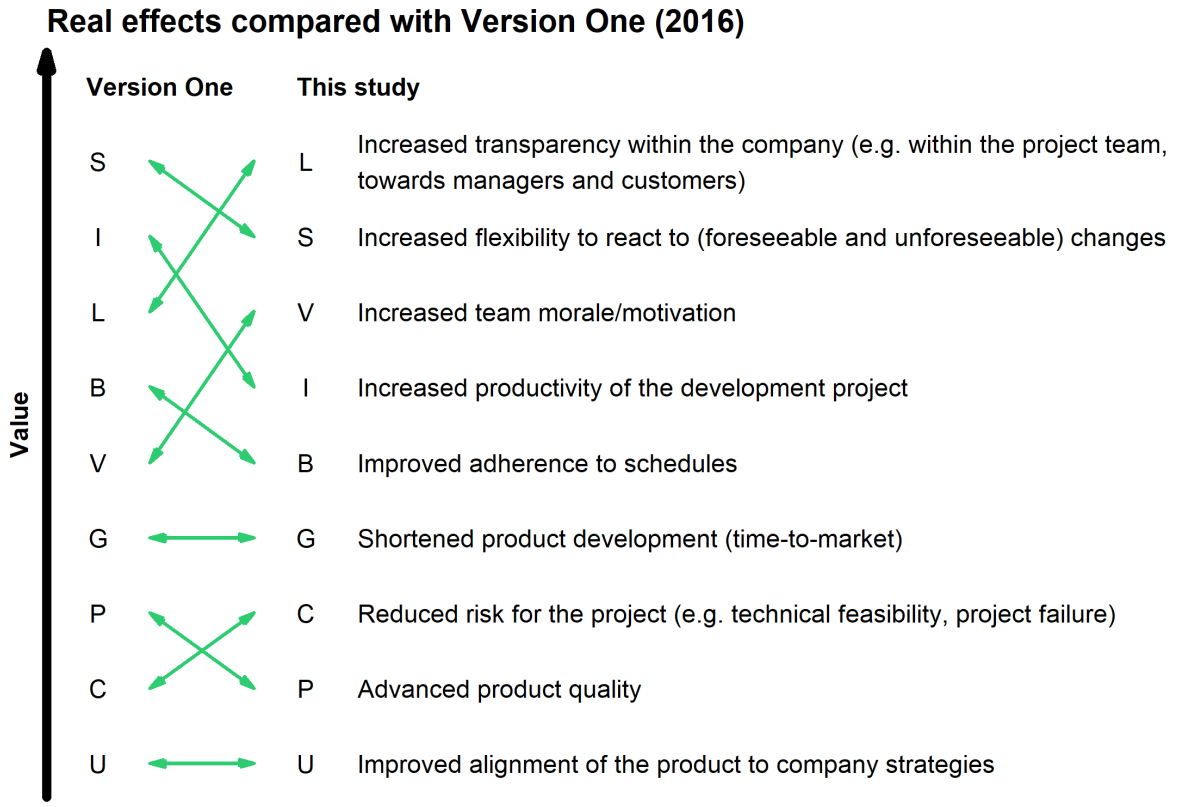


Figure 6.5.: Comparison of results from Version One (2016) and the study at hand.

Description
The diagram contrasts the results on real improvements derived by the well-known study of Version One (2016) and by the study at hand. While Version One focuses on agile software development, the study at hand investigates agile hardware development. As the list of partial values tested by Version One and the study at hand does not overlap completely, the diagram shows only those that were evaluated in both studies. The letter ID's used are randomly sorted and are consistent with other diagrams in this document (compare Table 6.1).

It is important to note that the survey of Version One asked the interviewees to select the partial values that agile software development has actually improved most (yes/no). In contrast, the study at hand asked the participants to rate each single partial value, to which degree it has in fact revealed as improvement (5-point Likert scale). However, the derived rankings of each study are comparable. This is what the diagram depicts. Minor differences in the rankings are marked green. Major differences ($>$ 3 ranks) are colored red.

Key learnings
- No major differences are visible.
- Agile development has indeed the same effects in both software and hardware industries.

Interpretation and interim conclusion
- Although there are slightly different motivations why to implement agile development in software or in hardware industries (compare Figure 5.5 and Figure 5.6), the real effects of agile development of software or hardware are very similar.

7. Applicability of Agile Development of Physical Products

After realizing that agile development is indeed beneficial in some industrial contexts, it remains interesting, if agility is limited to software development only. Thus, this chapter focuses on the overarching question whether and - if yes - to which degree agile development is applicable and valid in hardware development since the original concept (including agile methods and practices) was designed in and for the software industry.

Firstly, the study questions, if the well-known Manifesto of Agile Software Development (Beck et al. 2001) is transferable to agile hardware development. The Manifesto creates a philosophy built upon values and principles. By means of so-called agile methods and practices, the philosophy becomes practicable. If the Manifesto is valid in hardware development, too, the term "software" could be replaced consequently by "product" referring to "software and hardware".

Secondly, the author team collected challenges associated with becoming agile in product development from experience and literature. The interviewees were asked to evaluate to which degree they find presented challenges difficult to solve in practice. Similarly to Figure 6.1, the study contrasts expected challenges with actual challenges.

Analogously, the chapter continues by outlining additional costs and conflicts caused by agile hardware development. The former compares which monetary input is needed for organizing a team in an agile manner additionally to the costs caused by the existing development approach in responding companies. The latter refers to rather psychological challenges associated with the adoption of agile development.

Finally, the chapter concludes with interlinking certain time aspects. In this context, the author team was keen on how long companies are engaged in agile software and agile hardware development. Also cross references are identified between the implementation progress and the time companies are engaged in agile development.

7.1. Transferability of the Manifesto of Agile Software Development to the Development of Physical Products

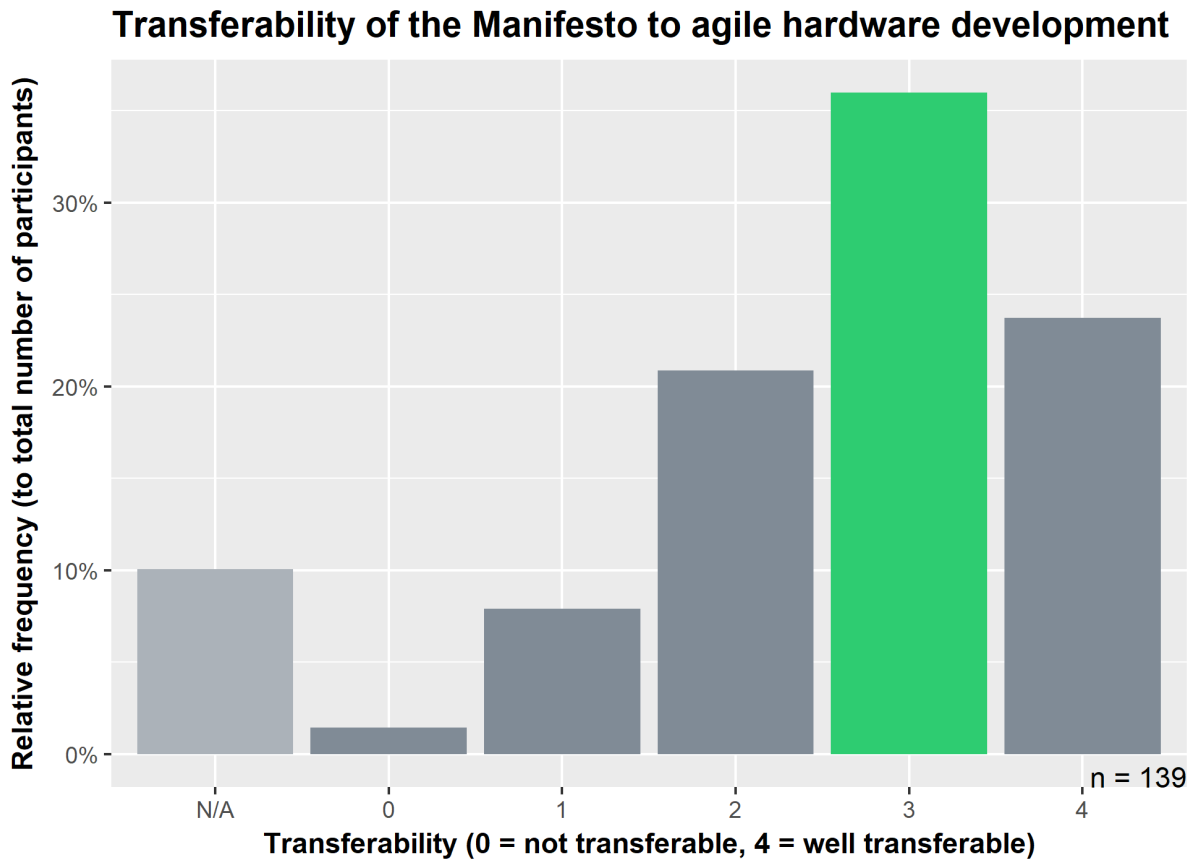


Figure 7.1.: Transferability of the Manifesto for agile software development to the agile development of physical products.

Description
Participants think that the Manifesto of agile software development is also applicable and valid for agile development of physical products. The distribution is left-skewed and reaches a mean of 2.8 (excl. N/A).

Key learnings

- Manifesto of agile software development is also valid for agile development of physical products.

Interpretation

- Agile development is a mindset predominantly, which is per se industry-independent (see also Figure 7.3.
- It remains questionable, if also agile methods and practices, that build upon the Manifesto, are transferable to agile hardware development.
- As a mindset is per se industry-independent, it might be possible that the Manifesto evaluation (asked in the survey) was mixed with a method or practices evaluation concerning the transferability to hardware development.

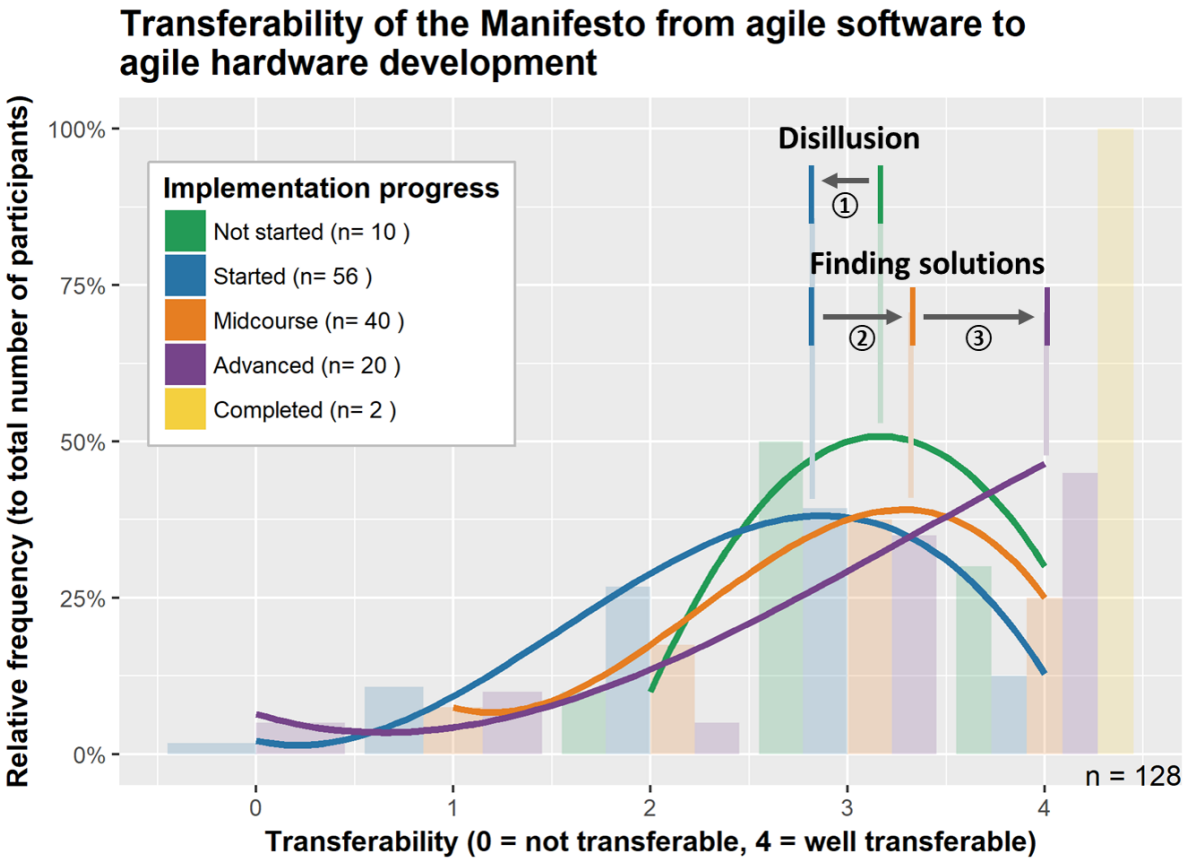


Figure 7.2.: Transferability of the Manifesto for agile software development to the agile development of physical products correlated with the implementation progress.

Description
Companies that have not yet started with the adoption of agile development think that the Manifesto is well transferable to agile hardware development. They are certain (small deviation) and have a high mean of 4.2 which is larger than the mean of those that just started (3.5, large deviation), those that are midcourse (3.9) and those that are advanced (4.1).

Among the participants only two said that they completely implemented agile development. Thus, the "completed" bar is shown only for the sake of completeness.

Key learnings

- The more advanced a company is in implementing agile development, the more the Manifesto is conceived to be valid in hardware development, too.
- Companies that have not started yet have high expectations, but seem to get disillusioned once started. During the time they find solutions and understand the core of agile development.

Interpretation

- The order of phases seem to follow Gartner's Hype Cycle. Initial euphoria and conviction leads to disillusion and finally to the slope of enlightenment and the plateau of productivity.
- Disillusion might stem from two reasons: (a) Adapting agile methods to company-specific context is not trivial, (b) as explained in Figure 6.2, some expectations on agile hardware development are inflated and turn out to be not realistic.
- The answers of experienced companies show, however, that agile development is possible in hardware industries, too, once initial hurdles are overcome.

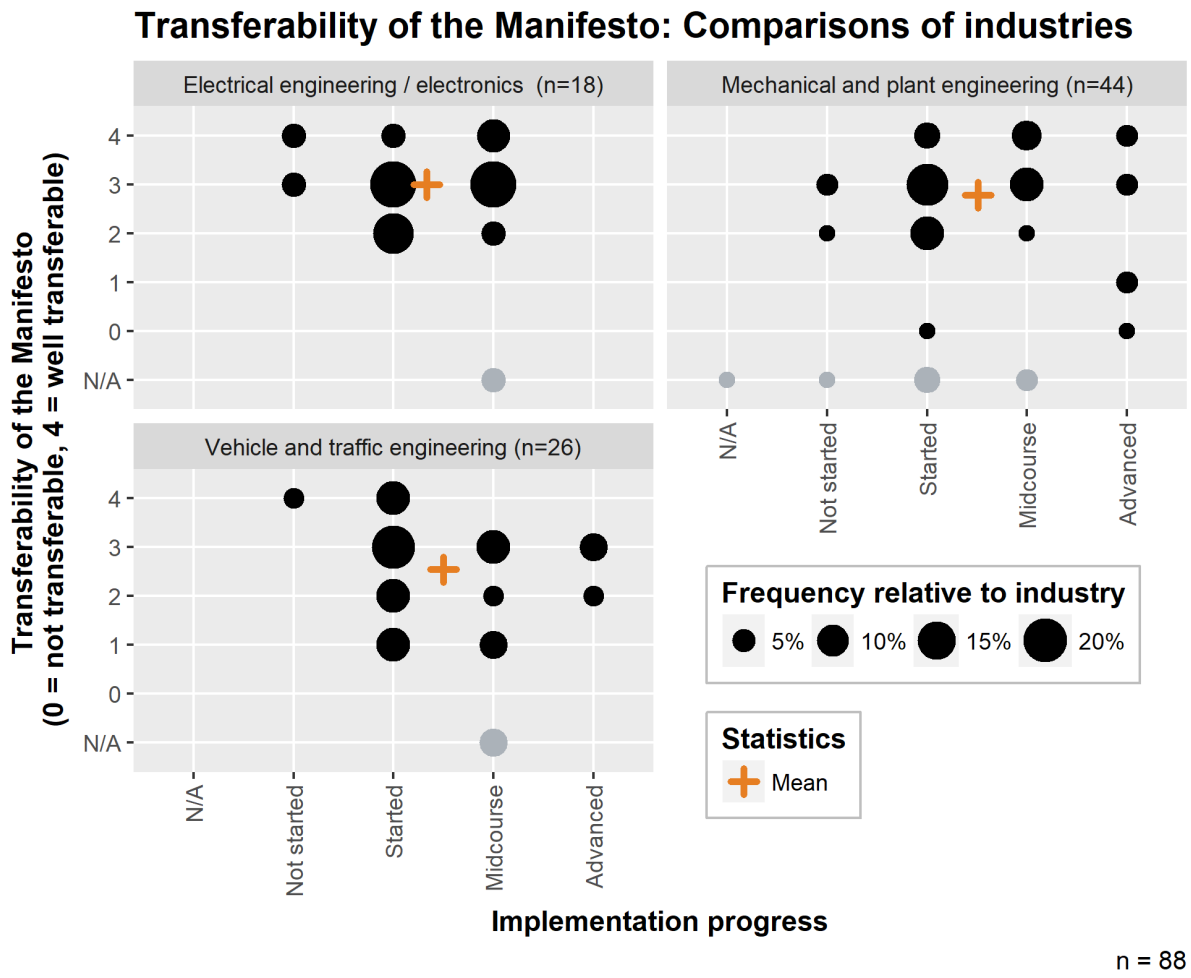


Figure 7.3.: Transferability of the Manifesto for agile software development to the agile development of physical products correlated with the implementation progress: Comparisons of industries.

Description
The circles depict the participants answers whereas the area symbolizes the frequency. The larger the area, the more participants chose the same rating. Frequencies are relative to presented industries. Hence, the sum of all circles' area within each industry equals 100% (excl. transferability N/A). The cross represents the balance point or mean respectively (excl. transferability N/A).

The transferability across presented industries does not differ a lot. However, electrical engineering / electronics industry is less dispersed and has a slightly higher balance point on the transferability dimension (although the total number of participants per industry is small).

Key learnings
- Industry affiliation does not seem to play a role for the transferability of the Manifesto.

Interpretation
- It might be easier to apply in industries that are either close to software or have a high software proportion (e.g. electronics).
- Adaption of agile methods and practices seem to be elaborate especially in vehicle and traffic engineering.

Interim Conclusion

The survey reveals that the Manifesto of Agile Software Development is transferable to agile hardware development, too. Hence, in the Manifesto, it is possible to replace the term "software" by "product" referring to "software and hardware". As explained in Figure 4.5), the Manifesto creates a philosophy. In Figure 7.3 the author team derives that this mindset is industry-independent.

However, agile hardware development is not trivial due to the constraints of physicality as it will be shown in Figure 7.6. Agile methods and practices were designed for software development. When companies start to become agile in hardware development, they apply agile methods and practices from the software industry (see Figure 4.12. Hence, reinterpretation and adaption to the circumstances and constraints of physical product development is required. Further research should be done to analyze cause-effect relationships for that and to investigate how to support companies in overcoming these challenges.

7.2. Challenges of Agile Development

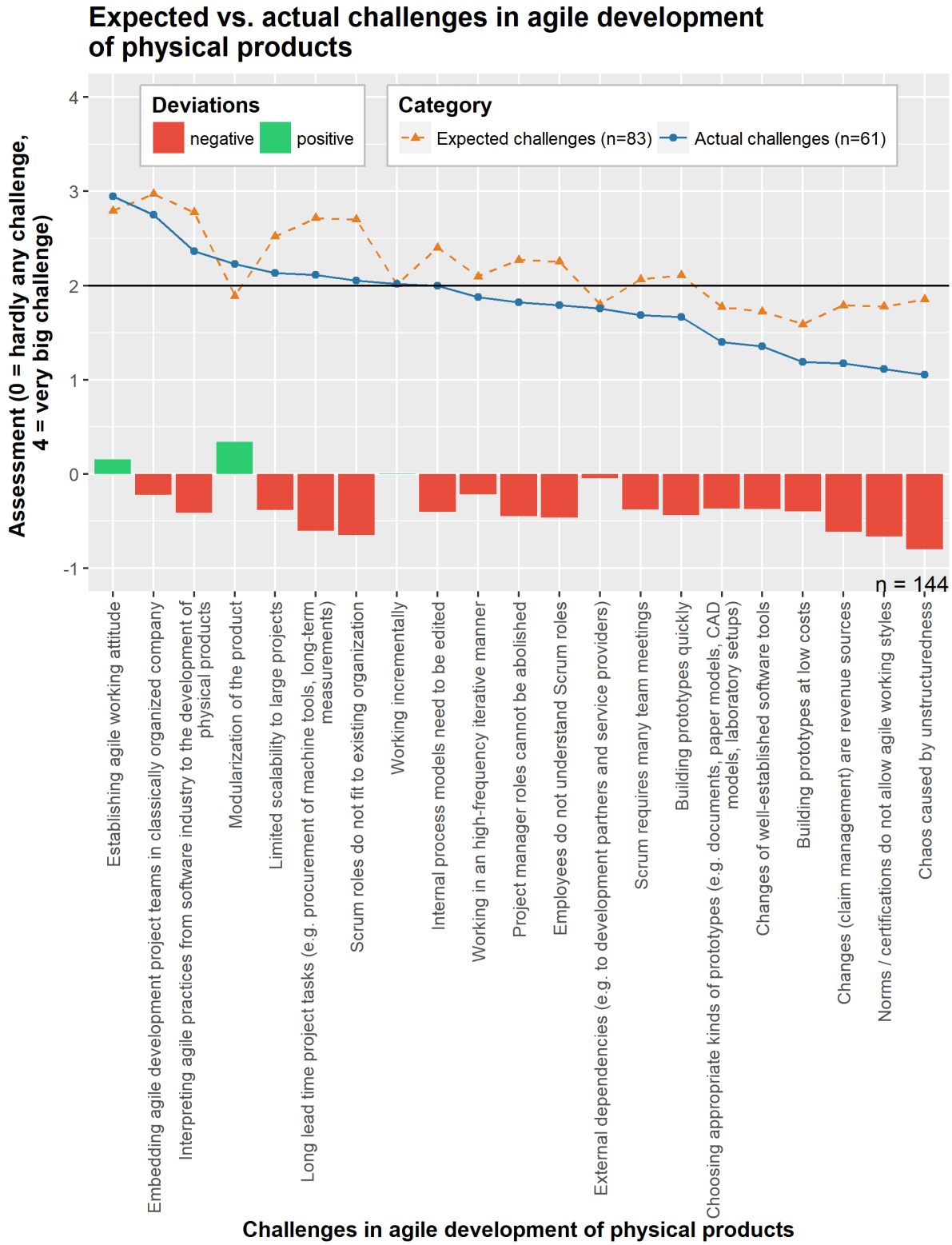


Figure 7.4.: Expected vs. actual challenges through agile development of physical products - sorted by actual challenges.

Description
The graph contrasts expected challenges with real challenges in agile hardware development. While the expected challenges graph includes all answers, the actual challenges graph excludes beginners. More precisely, the actual challenges graph considers only those participants that claim an implementation progress of "midcourse", "advanced", "completed" and "N/A". Please compare Figure 7.9 for the number of participants per implementation progress category.

Concerning the real challenges, establishing an appropriate mindset for agile development, embedding agile teams in classically organized companies and interpreting practices from agile software development are the Top 3. Chaos caused by unstructuredness, challenges caused by norms or certification requirements and issues due to claim management (Note: Some companies utilize changes as a reason to edit and increase the project price) are perceived as almost no challenge at all. These extrema have a very similar ranking when it comes to expected challenges.

However, the latter ones as well as "Scrum roles do not fit to the existing organization" deviate most. In general, participants expect a larger impact through investigated challenges in agile hardware development than they are in fact. Only product modularization and working incrementally is underestimated. Additionally, only a few investigated challenges reach a higher mean larger than 2 (which is the medium option of the scale).

Key learnings
- Among the tested challenges, most are overestimated. Companies expect larger impacts than they actually cause.
- The biggest challenges come along with implementing an appropriate mindset and nesting it into classically organized companies.

Interpretation and interim conclusion
- Many challenges result from organizational structures, that have been grown historically over decades in most companies, and the closely linked type of teamwork (collaboration vs. cooperation). The working style of agile development requires rather collaboration and decentralized decision making, that contradicts with cooperative and centralized decision making organizations.
- It seems to be apparent for the interviewees that adaption and interpretation of the agile methods and practice are mandatory for agile hardware development.

7.3. Additional Costs of Agile Development

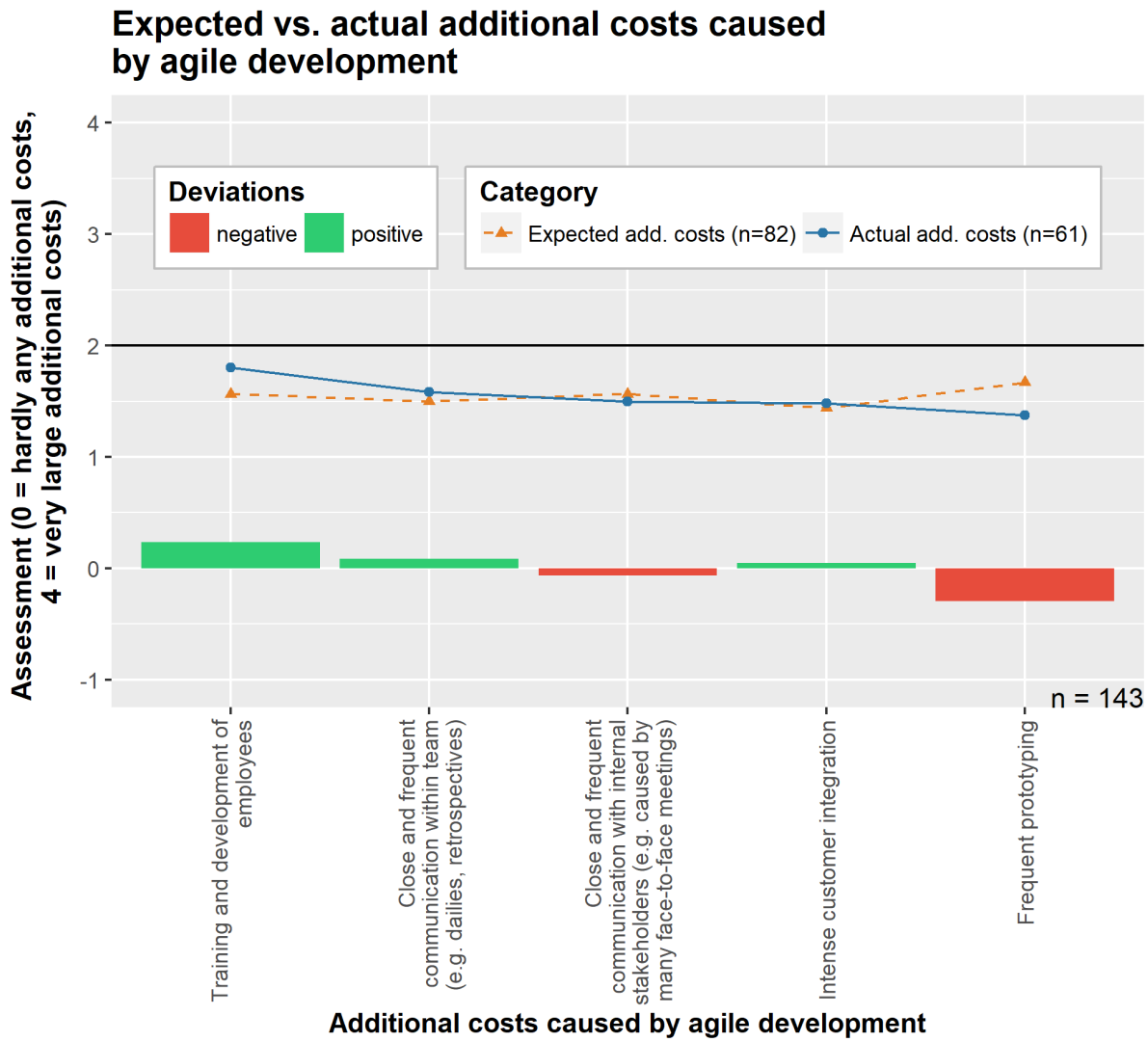


Figure 7.5.: Expected vs. actual additional costs through agile development compared to the development approach participated companies had before.

Description
The graph contrasts expected with real additional costs caused by agile hardware development. While the expected additional costs graph includes all answers, the actual additional costs graph excludes beginners. More precisely, the actual additional costs graph considers only those participants that claim a implementation progress of "midcourse", "advanced", "completed" and "N/A". Please compare Figure 7.9 for the number of participants per implementation progress category.

All additional costs provided are considered as quite marginal (<2). Expectations and real effects match well, except of "Frequent prototyping".

Key learnings
- Additional costs caused by agile development (at least among those aspects listed in the diagram) are considered marginal.

- Participants perceive the highest monetary effort in training and developing the employees which is even underestimated.
- Building frequent prototypes is not considered expensive.

Interpretation and interim conclusion

- As agile development is predominantly a mindset, training and development of employees is needed and very important.
- Agile development calls for potentially shippable increments in each iteration. Nevertheless, participants do not see much costs in prototyping physical products. This seems surprising as it is discussed as one of the most difficult challenge of agile hardware development in academic literature. It remains open what practitioners associate with the term "prototype" as in German "Prototyp" is often referred to a high fidelity mock-up.
- Although the listed additional costs correspond to highly rated challenges in Figure 7.6 (e.g. establishing agile working styles), they seem to cause almost no additional costs. Here, the survey data is inconsistent. On the one hand, changes in procedures and organizational structures cannot come without costs as agile development is a radical intervention in classical companies which even involves a mindset reframing. For such profound changes, it is very difficult to reorganize during every-day-business, instead additional resources are needed to implement the organizational change in a sustainable way. On the other hand, these changes require time and can not be accomplished within a couple of days. Furthermore, the effects of agile development can often be observed time-delayed. This reasoning leads to the conclusion that the interviewees might either not associate arising expenses with agility or they are not aware of them.

7.4. Conflicts Caused by Agile Development

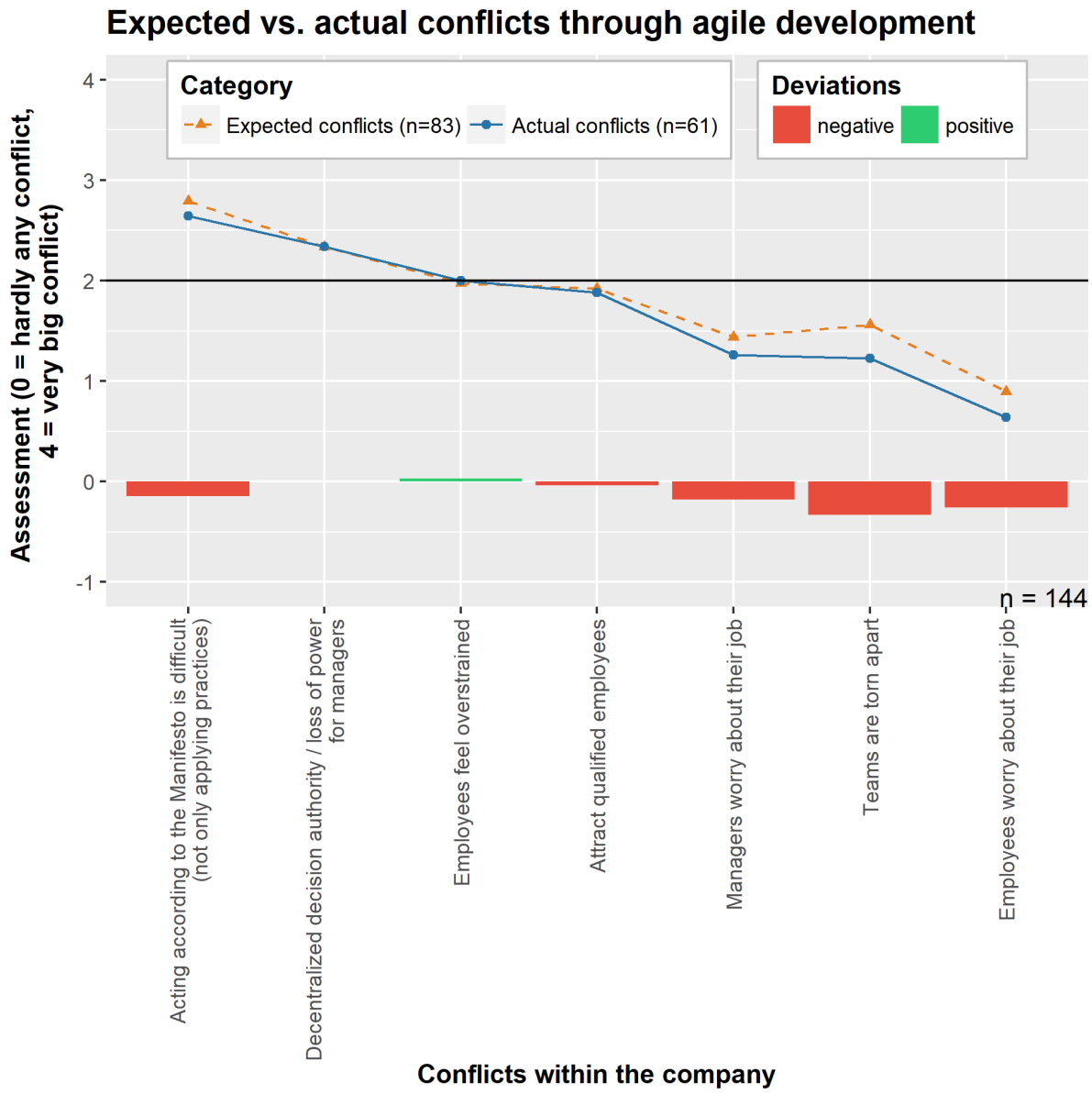


Figure 7.6.: Expected vs. actual conflicts through agile hardware development.

Description
The graph contrasts expected with real conflicts caused by agile hardware development. While the expected conflicts graph includes all answers, the actual conflicts graph excludes beginners. More precisely, the actual conflicts graph considers only those participants that claim a implementation progress of "midcourse", "advanced", "completed" and "N/A". Please compare Figure 7.9 for the number of participants per implementation progress category.

On the one hand, "Acting according to the Manifesto" (in line with the philosophy) and "Decentralized decision authority / loss of power for managers" cause conflicts most. On the other hand, job worries and team separation is perceived as less meaningful. Although overestimated, the difference between expected and actual values are small.

Key learnings

- Acting according to the Manifesto and loss of power cause conflicts within the companies.
- Job worries and team separation does not seem to cause serious conflicts.

Interpretation and interim conclusion

- On the personal level: Employees are maybe more willing to react to changes than expected.
- On the organizational level: As shown in Figure 4.1 and Figure 5.1 companies are confronted with dynamic context conditions otherwise they would not want to become more versatile or flexible. To implement agile development, modifications in their procedures and organizational structures are inevitable. However, the willingness to change on an organizational level does not seem to be large, or rigid organizations are hardly changeable on short notice.

7.5. Companies' Engagement with Agile Development

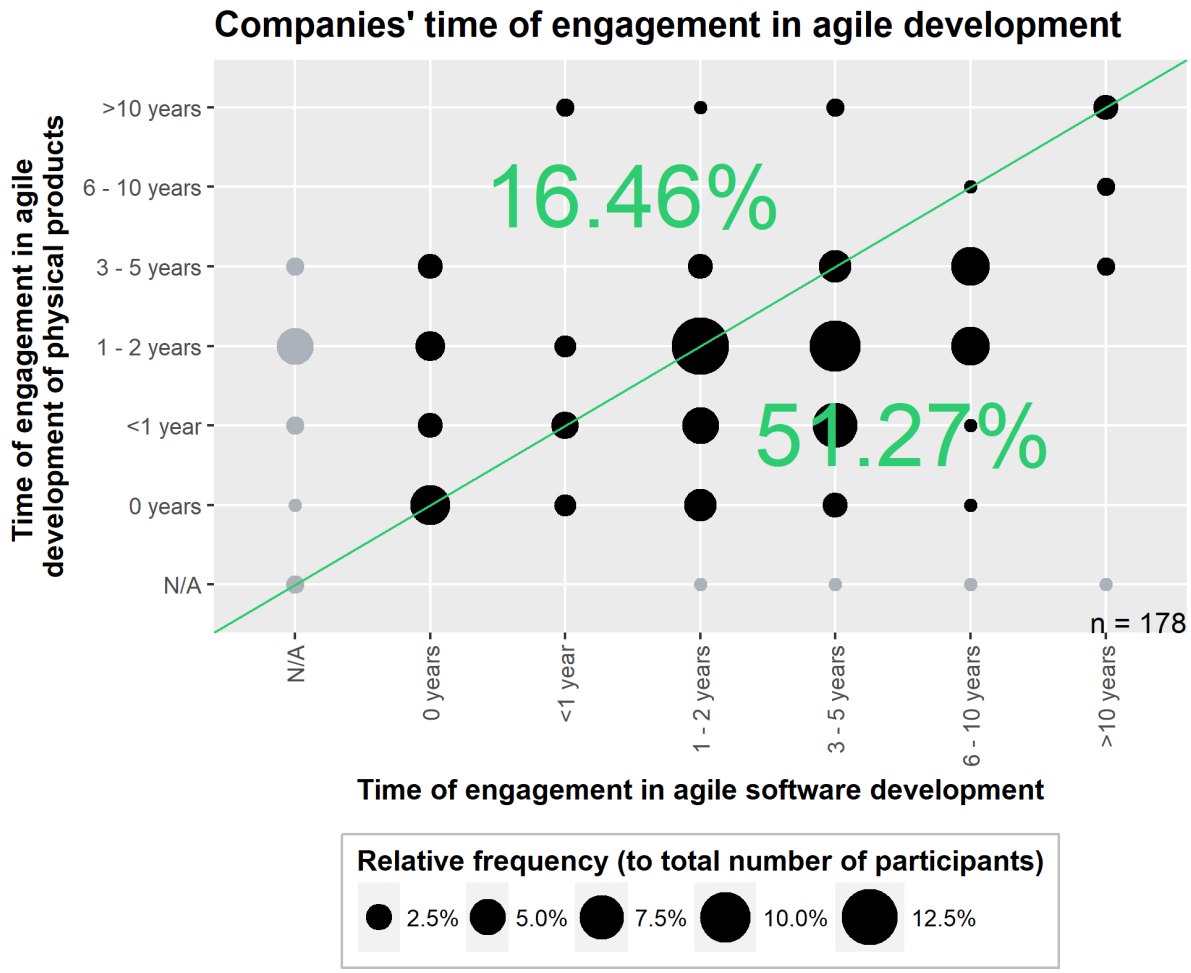


Figure 7.7.: Companies' time of engagement in agile software and agile hardware development.

Description
About 51% of participated companies are engaged longer in agile software development than in agile hardware development, while about one third started to implement agile development simultaneously in software and hardware development (being on the diagonal line). About 16% of the companies began to become agile in hardware development without experience in agile software development. Participants responding with N/A are excluded for the percentage calculation.

Key learnings
- Most companies start with agile software development before adopting it for hardware, too.
- Within the recent year, many companies started with agile hardware development without experience in agile software development.

Interpretation
- In most companies today, agile software development is an island in classically structured organizations. It can be considered as a separated department (silo) that is synchronized with neighboring departments such as hardware development. This is why agile develop-

ment in general might be not new for the company, but bringing the agile working style into hardware development is perceived a challenge (see Figure 7.4 as it is (a) a different development context (constraints of physicality) and (b) a new experience for employees in hardware development.

- Agile software development is a more mature concept since less critical challenges exist anymore because it has been applied in the software industry for about two decades. Starting with agile software development before trying it in hardware might come with lower risks.

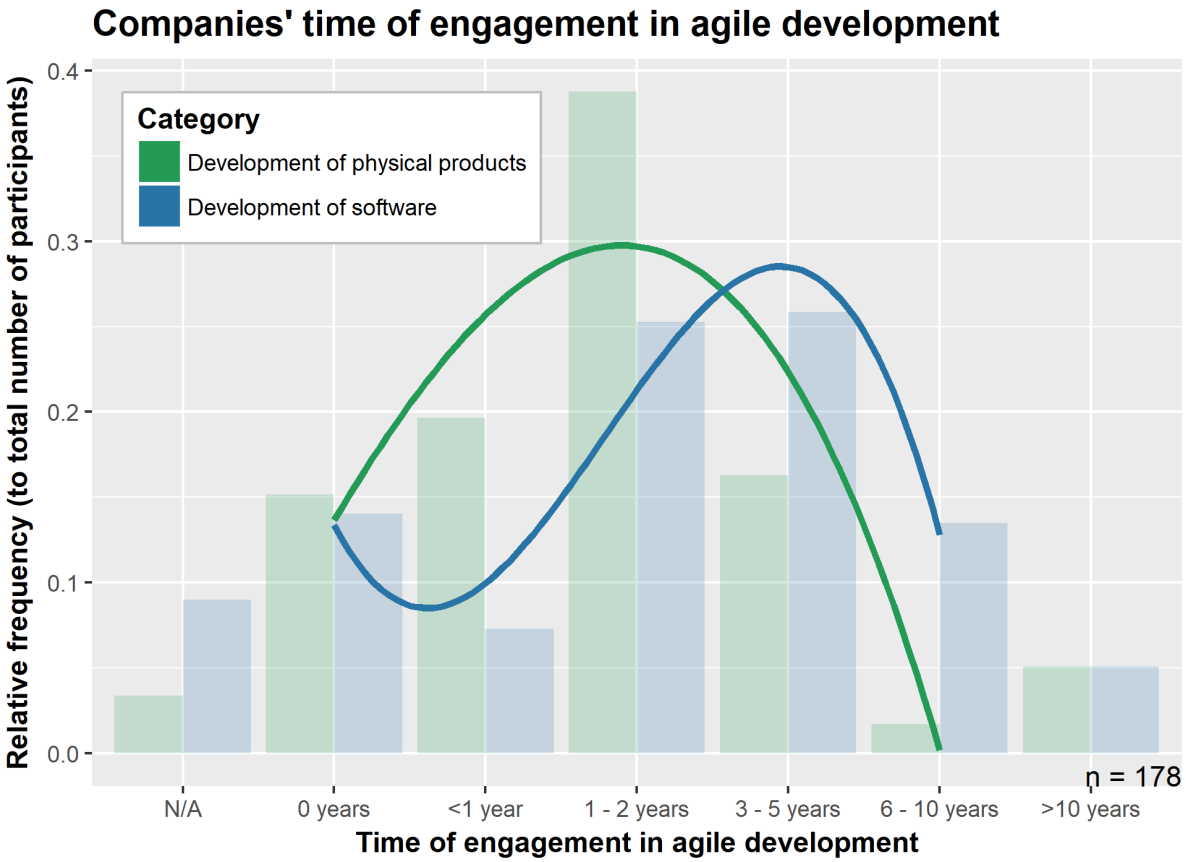


Figure 7.8.: Companies' time of engagement in agile software and agile hardware development.

Description
Companies have dealt with agile software development usually longer than with agile development of physical products. Remarkably, there is a cut in visibility for agile software development in the category "< 1 year". However, the 3rd degree polynomial regression lines form two waves whereas agile software development is about 1 to 5 years ahead.

Key learnings
- Companies that have dealt with agile hardware development have been engaged in agile software development for 1 to 5 years already.
- Most companies that have implemented agile development of physical products have dealt with it for 1 to 2 years.

Interpretation
- Agile hardware development is a very recent field of knowledge that is fed by agile software development in many aspects.

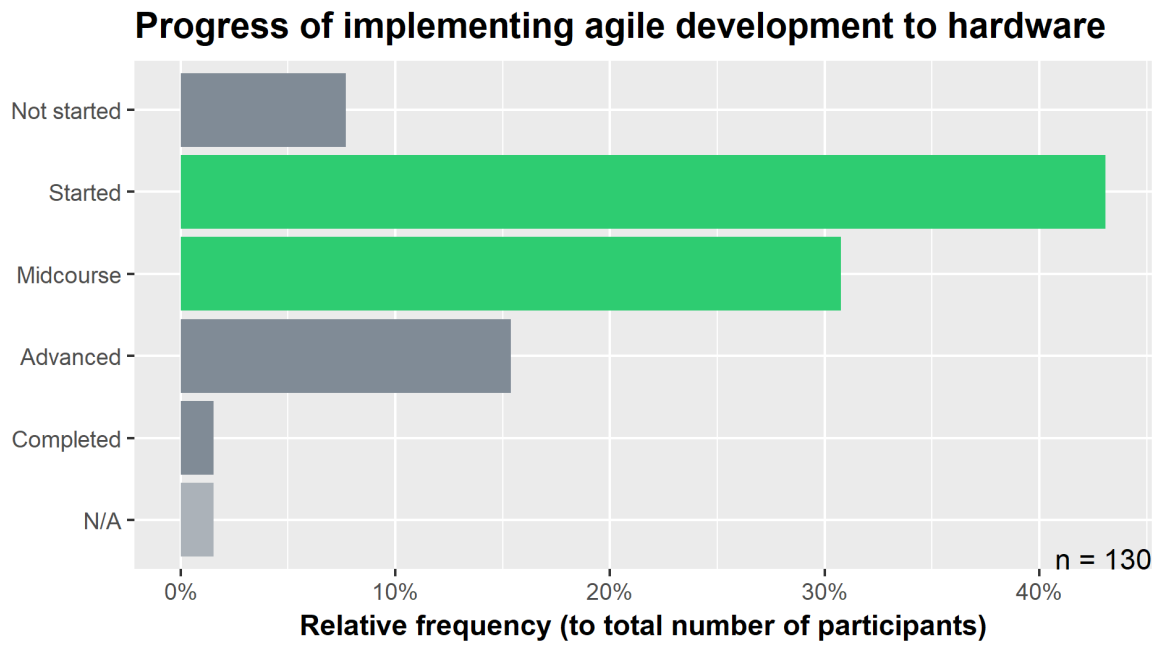


Figure 7.9.: Implementation progress of agile development of physical products.

Description
Companies from Germany, Austria and Switzerland just started implementing agile development to hardware projects. However, some companies that are much ahead of the bulk exist.

Key learnings
- Most companies have little experience in agile development of physical products.
- Some companies are much ahead of the bulk.

Interpretation
- Agile development of physical products in the industry is just about to gain momentum. It seems to be likely that learning curve effects across companies and industries drive the efficiency in implementation and agile method application.
- Imagining a trend line in the diagram, many companies will follow.

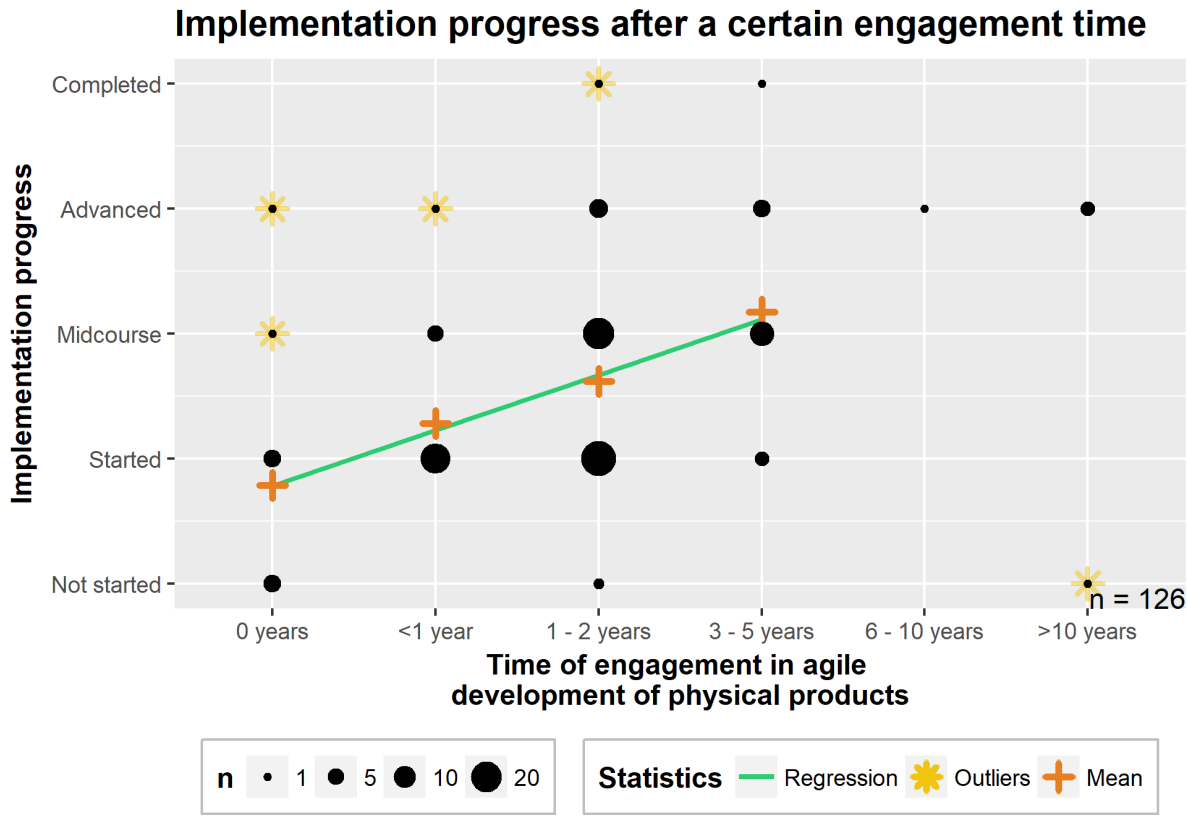


Figure 7.10.: Implementation progress after a certain time of engagement in agile hardware development.

Description
The diagram depicts the correlation between implementation progress and engagement time of the company with agile development of physical products. Dots represent given answers by the participants, orange crosses are the means per x-coordinate category and the green line stands for a linear regression. x-coordinate categories that contain insufficient data points (e.g. ">10 years") are excluded from regression analysis, but are drawn for the sake of completeness.

Obviously, the longer companies are engaged in agile hardware development, the more advanced they become. When companies have been engaged in agile development of physical products for 3 to 5 years, about 80% of the participants outstripped the starting phase. Furthermore, companies that stated that they have a midcourse implemention progress have been engaged in agile hardware development for at least 1 year (more than 80% chance).

However, some outliers exist. For instance, it does not seem practicable to have a midcourse or advanced implementation status at year 0. Those outliers are marked with yellow stars.

Key learnings
- The chance to achieve an advanced implementation level within 1 to 5 years is 50% since about 50% of the participants claiming an advanced progress said that they have been engaged 1 - 2 or 3 - 5 years already.
- Following the trend line, implementing agile hardware development needs probably more than 3 years.
- When companies say that they just started, they have dealt with agile hardware development for 2 years at maximum (75% chance).

Interpretation

- Implementation progress might be subjective as there might be no real ending point. Consequently, the implementation progress assessment might be fuzzy which leads to large deviations among the participants.
- Many companies are engaged in agile software development longer than in agile hardware development (see Figure 7.7). Possibly, outliers could have mixed the implementation progress with agile software development departments.
- Two participants claim that they completed the adoption of agile hardware development. It remains questionable, if this can be achievable within 1 to 5 years - if possible at all.
- Interestingly, many companies that have not yet started are engaged in agile hardware development for quite a long time. It remains open, if they have thought about adopting agile development or if they have tried it once, but failed and did not yet start over again.
- Agile hardware development is a recent field of knowledge. Necessary changes in organizational structures and procedures might have just been initiated in most companies.

Interim Conclusion

While in the software industry agile development is considered a standard approach, it is still a long way in hardware industries. However, agile hardware development is gaining momentum. Following the trend in Figure 7.9, many companies will start to implement agile development of physical products in the next years ahead. There is no turning point in sight, the author team rather anticipates that the visibility of agility (in development or in organizations in general) will increase exponentially.

Fed by the experience of agile software development, agile hardware development will probably reach maturity much faster. Nevertheless, developing hardware in an agile manner requires to also overcome the constraints of physicality (such as quick and cheap prototyping). This is an issue agile software development does not have to consider, thus it needs to be further investigated by research in order to support practitioners in increasing their agility in hardware development. As it was displayed in Figure 4.13, companies start to apply agile practices that support administrative project work and include design supporting practices later. Especially for the design supporting aspects of agile hardware development it is essential to learn to overcome the constraints of physicality.

Yet, it remains important to consider the "right" degree of agility. While a too high degree of agility can cause efficiency problems, an insufficient degree of agility leads to challenges in competing under VUCA conditions.

Part IV.

Conclusions

8. Summary and Final Remarks

Now that agile methods are successfully used in software development to deal with volatility, uncertainty, complexity and ambiguity (VUCA), these methods are increasingly being considered for the development of physical products. A dynamic development environment and the pressure to develop innovative solutions quickly pose great challenges for product development. Agile development promises efficient and effective solutions. At the same time, according to Gartner's hype cycle, agile development is still relatively immature for the use in the physical world, which has a significant impact on the motivation of application and thus on the importance and relevance of the methods (Schmidt, Weiss, and Paetzold 2018). Not only changed ways of thinking in project management, but also adapting agile methods to the specific characteristic of physical product development are necessary in order to avoid failures or to exploit the potential of the methods.

The present study provides a differentiated overview of the expectations and actual benefits of agile methods for the development of physical products. Three key findings can be derived from the study:

Motivations: Agile development has significant potential to make development processes more flexible, transparent and reactive. However, this does not necessarily mean that development lead times and costs are also reduced. Values and principles of agile development help to increase transparency, support self-organization, team morale and interdisciplinary cooperation, which not only helps to exploit opportunities but also increases the speed of reaction to high uncertainty and volatility. However, this is not necessarily related to reduced development lead times and costs. Although such a side-effect can occur, this is not the intention of using agile methods and should therefore not be used as a motivation to implement agile development.

Potentials: The question of whether agile development of physical products is actually hyped, that is, if the expectations in terms of benefit and efficiency are too high, needs to be considered in a more differentiated way. In fact, the study revealed that expectations were too high for "hard" controlling KPI's such as adherence to deadlines, productivity, and so on. On the other hand, expectations regarding the application of agile development in rather "soft" parameters such as transparency, optimization of learning processes and motivation to work were clearly exceeded. It is therefore important to define adequate evaluation criteria based on the actual objectives of the application of agile development.

Applicability: The values and principles according to the Manifesto that are pursued with agile software development are industry-independent and are also accepted in physical product development. However, a method adaptation seems necessary in order to be able to deal with the specific differences and particularities, especially in interdisciplinary development tasks for mechatronic or cyber-physical systems. However, it could be shown that an understanding of this circumstance is more pronounced with an advanced implementation step of agile development compared to where the implementation is just beginning or where it is being prepared.

It can be concluded that agile development of physical products provides significant advantages

for solving complex development tasks. But there is a need for action to adapt agile methods to the specifics of physical products. This includes not only physical challenges (prototypes) but also the scalability.

Exaggerated expectations, as identified in the study, also entail the risk that - if goals are not achieved - the application of the method will be discarded. This would neglect a considerable potential for product development. Regarding this, the editors hope that this study will not only clarify the performance potential of agile methods for physical product development with the differentiated consideration of benefits and expectations, but will also provide indications for the introduction, adaptation and application of agile methods in the sense of a beginning synthesis.

References

Schmidt, Tobias Sebastian, Stefan Weiss, and Kristin Paetzold (2018). "Expected vs. Real Effects of Agile Development of Physical Products - Apportioning the Hype". In: *International Design Conference (Design'18)*. Dubrovnik: to be published in May.

Beck, Kent, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas (2001). *Manifesto for Agile Software Development*. URL: `agilemanifesto.org`.

Version One (2016). *10th annual State of Agile Report*. VersionOne Inc.

Komus, Ayelt, Moritz Kuberg, Armend Arifaj, Christopher Bach, Christopher Berg, Dominik Czeicke, Andreas Fritzler, Burhan Hamziu, Sebastian Jack, Christoph Janowski, Patrick Kengji, Jens Schäfer, Matthias Schulz, Jason Setiabudi, Pascal Sonntag, Stefan Sprenger, and Anuphong Wongwat (2017). *Abschlussbericht: Status Quo Agile 2016/2017 (für Studienteilnehmer)*. Koblenz: University of Applied Sciences Koblenz.

About the Organizations

Institute of Technical Product Development: The research activities at ITPE focus on mastering complex socio-technical systems. For the description of complex technical systems, methods for the context-specific use of MBSE must be developed, which represent both product structures and process-driven information flows in the development processes. At the same time, classical process analysis approaches are coupled with methods of network theory in order to understand and analyze data and information flows within the company. The method linkage serves not only to optimize IT structures but also to support communication and collaboration processes in development. The increasingly dynamic environment for development makes it necessary to embed stronger aspects of adaptability and flexibility in product development. For this, agile methods prove to be an adequate solution and are therefore the focus of research activities.

AGENSIS Management Consultants: Founded 2002 in Munich (Germany), AGENSIS is dedicated to optimize product development systems and R&D Management. The customer portfolio built up over the years includes both global players and mid-sized companies, each with a strong technology and engineering focus. AGENSIS supports its clients with its unique knowledge and a specialized tool-set based on the lean development and agile development concepts and methods. Our experienced teams cover and support all domains of mechanic, mechatronic and software development and the corresponding process areas of project management, purchasing, sales and manufacturing. In our projects we advise and accompany both the workforce and the management of our clients towards the analysis of the status quo and the change transition to a more efficient and high competitive product innovator and technology leader. Aiming at excellence is our stimulus to focus our clients on sustainable values in product development.

Agile Development of Physical Products

More and more companies feel confronted with rising volatility, uncertainty, complexity and ambiguity (VUCA) in their development of physical products. To encounter such environments, they try to become more agile. However, many myths, misunderstandings and misinterpretations exist in agile hardware development. This empirical study sheds light on companies' motivations (reasons to implement agile hardware development), potentials (real improvements through agile hardware development) and the concept's applicability in hardware development. It provides quantitative facts by means of scientific methods.

In collaboration with:

Institut für Technische Produktentwicklung

AGENSIS MANAGEMENT CONSULTANTS

Universität der Bundeswehr München

ISBN 978-3-943207-28-6