

# On the Use of Hybrid Development Approaches in Software and Systems Development: Construction and Test of the HELENA Survey

Marco Kuhrmann<sup>1</sup>, Jürgen Münch<sup>2</sup>, Philipp Diebold<sup>3</sup>, Oliver Linssen<sup>4</sup> und Christian R. Prause<sup>5</sup>

**Abstract:** A software process is the game plan to organize project teams and run projects. Yet, it still is a challenge to select the appropriate development approach for the respective context. A multitude of development approaches compete for the users' favor, but there is no silver bullet serving all possible setups. Moreover, recent research as well as experience from practice shows companies utilizing different development approaches to assemble the best-fitting approach for the respective company: a more traditional process provides the basic framework to serve the organization, while project teams embody this framework with more agile (and/or lean) practices to keep their flexibility. The paper at hand provides insights into the HELENA study with which we aim to investigate the use of "Hybrid dEveLopmENt Approaches in software systems development". We present the survey design and initial findings from the survey's test runs. Furthermore, we outline the next steps towards the full survey.

**Keywords:** software process, development, empirical study, survey

## 1 Introduction

Different project contexts require adequate development approaches to properly address the various challenges in software projects. As major game plan, software processes are defined to capture all relevant knowledge and concepts required to provide guidance and templates to project managers and developers alike. For several years, we find a growing number of development approaches addressing different contexts and each emphasizing different aspects, such as strict guidelines, planning support, extensive templates, team behavior, or empowerment of the individual team members. Currently, we find two major streams in software processes: traditional (also referred to as plan-driven or rich) and agile processes. Also, there is a trend towards "agilizing" software (and system) development, since traditional processes are considered "heavy-weight" thus limiting the team's space of action and creativity as well as project speed. Yet, in recent research, we find strong indication that neither the pure traditional nor the pure agile approach are (exclusively) implemented in practice. Apparently, a pattern called "Water-Scrum-Fall" [We11] seems to provide the means for companies to define and implement context-specific development approaches. Initial research focusing on this phenomenon [Th15] provided first evidence

---

<sup>1</sup> University of Southern Denmark, Mærsk Mc-Kinney Møller Institute, Odense, kuhrmann@acm.org

<sup>2</sup> Reutlingen University & University of Helsinki, juergen.muench@reutlingen-university.de

<sup>3</sup> Fraunhofer IESE, Process Engineering, philipp.diebold@iese.fhg.de

<sup>4</sup> FOM Hochschule für Oekonomie & Management, oliver.linssen@fom.de

<sup>5</sup> Deutsches Zentrum für Luft- und Raumfahrt (DLR), Raumfahrtmanagement, christian.prause@dlr.de

that the “Water-Scrum-Fall” has become reality. However, recent research lacks quantitative data to eventually conclude the reasons for using so-called *hybrid development approaches* and to work out, which combinations can best address a specific situation.

**Problem Statement.** While agile methods are considered the most promising route towards faster development of high-quality software systems, many companies still use traditional software processes for several reasons. Different independently conducted studies, such as [KL15, VB15], show hybrid-method approaches implemented by those companies, especially when they have to deal with software systems in regulated domains. In [Th15], we conducted a systematic literature review to get an overview of the quantitative and qualitative data regarding the use and the combination of the different processes. A major finding is that, notably, quantitative numbers are missing and, therefore, an only incomplete picture regarding the use of the different development approaches is drawn. This particularly hampers the ability of process engineers to design context-specific development approaches grounded in evidence.

**Objective, Method, and Contribution.** To overcome the aforementioned situation, we designed a study with the purpose of working out (1) which development approaches are used in practice, (2) how and why are these approaches combined with each other, and (3) how standards, norms and regulations impact the design and implementation of agile methods in practice. Our study is designed as a mixed-method research approach in which we combine online questionnaires and interviews. The first part of the study is HELENA (Hybrid dEveLopmENt Approaches in software systems development), an online survey that helps the general (quantitative) data collection. The second part comprises a number of interviews. The paper at hand provides an overview of the entire study, the construction procedure of the HELENA instrument, initial findings from HELENA’s trial runs and, finally, an outlook concerning the next steps.

**Outline.** The remainder of this paper is structured as follows: In Section 2, we briefly summarize selected related work. Section 3 provides an overview of the overall research and delivers details on the HELENA survey in particular. Initial results of HELENA’s trial runs are presented in Section 4, before we conclude the paper in Section 5.

## 2 Related Work

The research presented aims to shed light on the use of traditional and agile development approaches and combinations thereof. Empirical studies providing quantitative data are rare, for instance, [FK07, KF15, KL15, Th15, VB15]. Furthermore, most of the studies available (we exemplarily mention the annual VersionOne study series [VerOne]) focus on the use of agile methods exclusively rather than drawing an integrated picture of the process ecosystem as such, e.g., [DPL16]. As we can also see from [Din12], research on agile methods is dramatically increasing, while research on traditional approaches is (more or less) stagnating. A full discussion of studies on process use can be taken from our previously published article [Th15]. The research presented in the present paper aims at filling a gap in literature by presenting an instrument and initial quantitative results on the process use. It also lays the foundation for future work.

### 3 The Overall Design of the HELENA Survey

Our research is organized as a mixed-method study comprising two parts. In this section, we provide an overview of the overall study design. In particular, we provide details on the HELENA survey from which we present initial results (from the survey's trials).

#### 3.1 Overall Study Design

The overall research approach is shown in Figure 1. The study design follows a multi-staged mixed-method approach, which is grounded in a number of previously conducted studies. In particular, parts of the presented research emerge from the IOSE<sup>2</sup>W survey [FK07], which is also the ancestor of the *3ProcSurvey* [KF15].

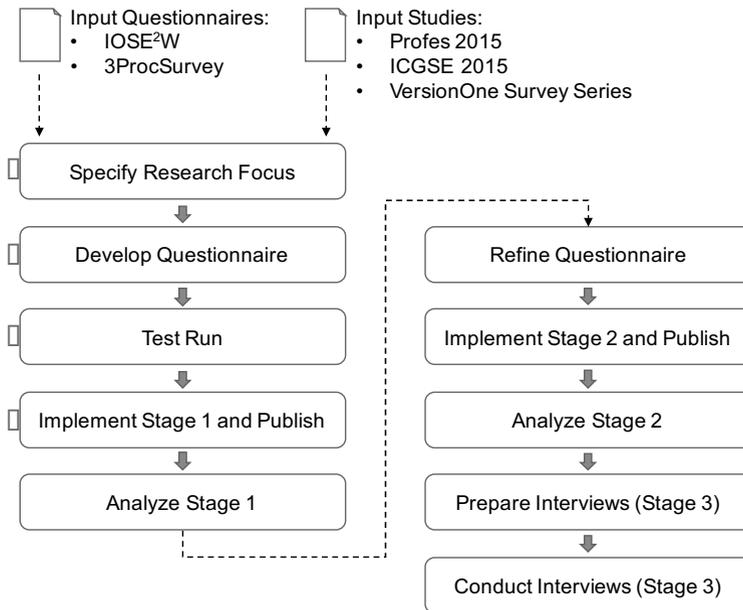


Figure 1 General research approach around the HELENA survey project.

Both studies [FK07, KF15] aimed at investigating how companies conduct their software development and process improvement. These studies served as the prototypes to derive the survey questions (in fact, several questions are still the same or, at least, developed in a similar manner to allow for comparison of data collected over time). Furthermore, the presented research is motivated by some discussion around our previously published paper [Th15], which aimed at collecting quantitative data on the use of the different development approaches (based on studies obtained from a systematic review and data from the annual VersionOne survey [VerOne]). The major finding that such data is, if at all, only scarcely to find is actually the main driver behind HELENA.

Based on these inputs, we developed the research design illustrated in Figure 1, which we describe in the following. The overall research approach comprises three stages:

1. In the first stage, we use an online survey to obtain quantitative and qualitative data.
2. Based on the analysis of the data obtained in Stage 1, in Stage 2, we plan to refine the questionnaire instrument and to conduct another, yet better scoped, data collection.
3. The results from Stage 2 form the input for Stage 3 in which we plan to conduct interviews with selected participants to clarify/confirm findings.

The research is grounded in our finding from [Th15] that the “Water-Scrum-Fall” has become reality. This finding defines our working hypothesis and helps specifying the research focus (Step 1) as follows:

*We accept that the “Water-Scrum-Fall” has become reality and that hybrid development approaches shape today’s software development landscape.* Hence, we aim at studying the following main questions:

1. Which development approaches are used in practice?
2. How are the development approaches used in combination?
3. Who makes the decision whether or not and how to apply a particular combination of development approaches?
4. Do application domains and systems’ complexity, the respective standards, norms, and regulations affect the construction of hybrid models?
5. How are hybrid development approaches constructed, maintained and improved?
6. What is the role of agility in the software and systems development landscape?

Complementing, the research also aims at investigating experiences regarding the use of different approaches in the respective domains.

In the second step (Figure 1), we iteratively developed the questionnaire for Stage 1 (see Section 3.2). Afterwards, we conducted a trial in collaboration with the DLR and FOM (Section 4), before we released the Stage 1 questionnaire in the beginning of May 2016. Stage 1 was open for answers until July 20, 2016, where we started Step 5 (analysis of the results from Stage 1).

The second stage will start after the evaluation of the data obtained in Stage 1, which might result in a revision of the questionnaire. In the second stage, a more focused data collection (potentially as follow-up of Stage 1) will be conducted to improve data quality and general reliability of the data. Finally, in the third stage, several interviews are planned to confirm findings and get further insights into practitioners’ ways of work.

Currently, our research is in Stage 1 in which we focus on conducting and analyzing the first HELENA survey instance. In the following, we present insights into the survey, its construction, and report initial findings and lessons learned obtained from the trial runs.

### 3.2 The HELENA Survey

The first part of the study is the survey “Hybrid dEveLopmENt Approaches in software systems development” (HELENA). The purpose of the questionnaire is to carry out an international data collection concerning the use and the combination of the different development approaches. The questionnaire used for the survey contains 25 (basic) questions, which are summarized in Tab 1<sup>6</sup>.

HELENA was iteratively developed and (externally) tested twice. We report the initial findings in Section 4. In the first iteration we used Google Forms to implement the questionnaire and released it. The trials, however, were carried out using a paper-based version of the questionnaire. To make these data available, we created a copy of the main survey from, which is available for internal use only, and entered the returned questionnaires manually into the data tables.

## 4 Selected Initial Findings from the Trials

This section reports selected findings from the two external trial runs. One trial was conducted at a company-wide, internal workshop of software developers from different branches of the German Aerospace Center (DLR). The second trial was carried out as part of the teaching activities at the FOM. The purpose of the trials was to test the instrument as such, i.e., clearness of the questions and timing. The purpose was **not** to create data for the actual data analysis, yet, we also use the data obtained to test the planned evaluation instruments. For space limitations, in this paper, we only use descriptive statistics (i.e., charts and tables) to present a brief overview of the data, since we consider these early data still informative.

No.	Group	Question
1.	M	What is your organization's size?
2.	M	What is the main business area of your company?
3.	M	Do you participate in distributed software projects?
4.	M	In which country are you personally located?
5.	M	In which application domain are you most frequently involved?
6.	M	Which role are you most frequently assigned to?
7.	M	In your projects, a software failure potentially can: [...]
8.	PU	Does your company define a company-wide standard process for software and system development?
9.	PU	Which of the following development approaches and practices do you use?
10.	PU	Do you combine different development approaches?
11.	PU	For the following standard activities in your projects, please indicate to which degree you carry out these activities in a more traditional or more agile manner.
12.	PU	What is the main motivation for this particular combination of development approaches?

<sup>6</sup> Note that the actual question structure can slightly differ when implemented in the respective survey tools. For detailed information, for HELENA, a *codebook* is available comprising detailed information regarding questions, conditions, value ranges, and variables. The codebook is available on request.

No.	Group	Question
13.	PU	How were the combinations of development approaches in your company developed?
14.	PU	How do you select your project-specific development approach?
15.	PUS	Which external standards are implemented in your company?
16.	PUS	Why have you implemented the aforementioned standards?
17.	PUS	How is the compliance of the development process assessed?
18.	PUS	Does agility challenge the implementation of the standards you have to apply?
19.	PUL	Is your development approach continuously improved?
20.	PUL	What is your motivation for implementing an improvement program?
21.	PUL	Is your company, unit or project certified?
22.	PUL	What are the goals of your improvement program?
23.	E	Based on your personal experience, please rate the following statements:
24.	E	Based on your personal experience, can you name problems occurred regarding your current process and your current application domain?
25.	Closing	Do you have any further comments or issues not addressed so far?

Tab 1: Overview of HELENA's basic questions. The questions are ordered in the following groups: M=metadata, PU=process use, PUS=process use and standards, PUL=process use and lifecycle, E=experience

#### 4.1 Population and Demographics

Our trials were carried out at DLR (11 responses) and FOM (4 responses), which makes 15 responses in total. Respondents stated that they were acting in the following roles: project manager (1), architect (2), developer (6), and miscellaneous (6), whereas the latter ones mentioned to be in different roles depending on the project and the respective activities. All interviewees are practitioners, even the FOM students, who are experienced practitioners working in industry and conducting their studies in parallel.

#### 4.2 Use of Development Approaches

To provide an impression of the kind of data obtained from the HELENA survey and the different options of evaluating and interpreting these data, in the following, we provide some selected data points. Please note that the data presented herein is not used for any interpretation, as it still is the test data. However, this data shows the general direction of the data, and, at the same time, our approach to answer the aforementioned questions.

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	Total
Agile Portfolio Management (APM)																0
Behavior-driven Development (BDD)	1				1											2
Code Review	1	1			1	1			1	1	1	1	1	1	1	11
Coding Standards	1				1	1	1	1		1	1		1	1	1	10
Collective Code Ownership					1											1
Continuous Deployment									1		1		1	1	1	5
Continuous Integration	1				1	1		1	1		1		1	1	1	9
Crystal Family																0
Daily Standups								1								1
Definition of Done	1				1			1		1						4
Definition of Ready	1									1						2
DevOps								1	1							2
Digital Taskboard											1					1
Disciplined Agile Delivery (DAD)																0
Expert-/Team-based estimation (e.g., Planning Poker)							1	1								2
eXtreme Programming (XP)	1											1				2
Feature-Driven Development (FDD)		1								1		1				3
Formal Estimation (e.g., COCOMO, FP)																0
Formal Specification						1										1
Iteration Planning	1	1			1											3
Iterative Development	1				1	1								1		4
Kanban											1				1	2
Large-Scale Scrum (LeSS)																0
Lean Development							1				1					2
On-Site Customer																0
Pair Programming					1									1		2
Prototyping										1		1		1	1	4
Rational Unified Process (custom variant)				1												1
Rational Unified Process (standard version)																0
Refactoring		1			1	1	1		1		1		1	1	1	9
Release Planning		1		1	1						1			1		5
Retrospectives	1				1			1								3
Scaled Agile Framework (SAFe)																0
Scrum	1	1						1	1			1		1	1	7
Spiral Model																1
Test-Driven Development (TDD)	1				1			1							1	4
Unit Testing	1	1			1	1		1	1	1		1	1		1	10
V-Model Derivate(s)														1	1	2
Waterfall/Phase Model	1							1				1			1	4
Other			1		1											2

Figure 2 Overview of the responses regarding the development approaches used.

Figure 2 provides an overview of the responses for question 9 (*Which of the following development approaches and practices do you use?*; cf. Tab 1). The test data shows six development approaches being favored. In particular, code reviews, coding standards, continuous integration, refactoring, Scrum, and unit testing are widely applied. Furthermore, Figure 2 shows how these approaches are combined with each other. For example, R14 mentions to combine a V-Model derivate with Scrum, and R1 mentions a “classic” Waterfall/phase model being combined with Scrum; both are representatives of West’s “Water-Scrum-Fall” [WE11] as already found in [Th15]. Furthermore, again, we see the Rational Unified Process (RUP) in its standard version not applied at all (consistent with our findings from [KF15]). For R2, we see Scrum combined (or refined) with unit testing, refactoring, or Feature Driven Development (FDD), which also confirms a finding from Diebold et al. [Die15], that practitioners do not even use Scrum by the letters. Therefore, even though we only inspect test data, still, we can find trends observed in previously conducted research, i.e., trends to be also investigated in future work.

### 4.3 Traditional or Agile, or Both?

Another aspect of interest is the focus of the methods applied, i.e., which parts of a project are run rather agile or traditional, or balanced. For this, question 11 (*For the following standard activities in your projects, please indicate to which degree you carry out these activities in a more traditional or more agile manner*; Tab 1) aims at working out for which activities people opt for (more) traditional or agile approaches.

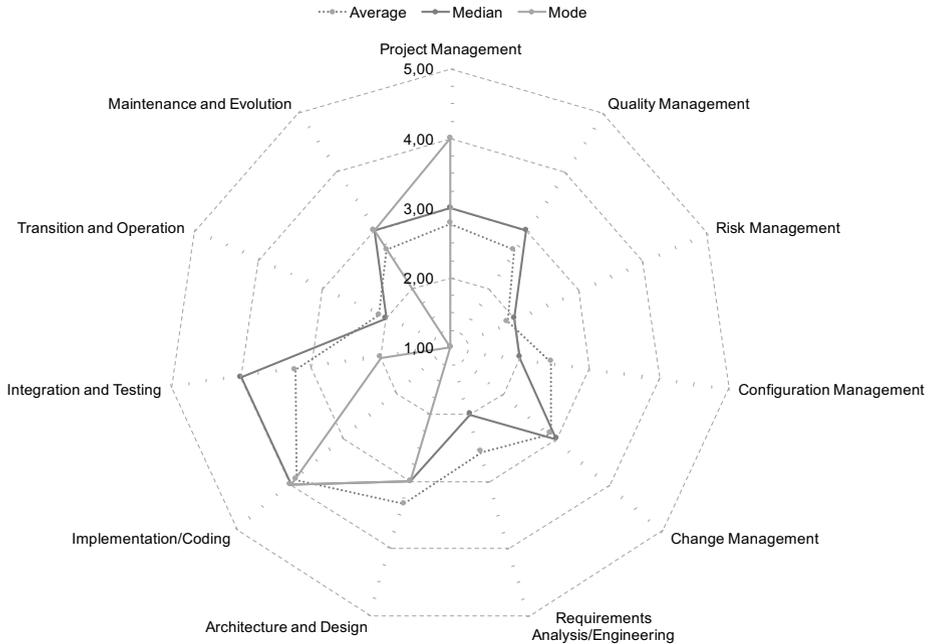


Figure 3 Overview of the implementation of agility and traditional processes in different software engineering disciplines in respondents' projects. The Likert scale used reads as follows: 1=fully traditional, 3=balanced implementation, 5=fully agile implementation.

Figure 3 provides an overview of the results from the trials as a radar chart. Please note, in the trials, we received three invalid data points for this question, i.e., the chart is based on 12 valid answers only. To present the rating, we use the average, median, and mode values to illustrate the findings. The test data shows that project management is implemented fairly balanced (with a slight tendency towards more agility). Risk, and configuration management as well as transition and operation show the exact opposite trend. Yet, implementation and coding, integration and test, and also architecture and design are to a large extent implemented in an agile fashion.

#### 4.4 Agility and Standardization

One of the main questions we want to investigate with HELENA is whether agility challenges the implementation of standards, especially in regulated domains. For this particular question 18 (Tab 1), we received nine valid answers of which eight stated “no, agility is easy to implement in our context”, and only one respondent stated “yes, agility challenges the implementation”. The reason for the latter statement was: “*bureaucratic standards reduce speed in projects*”.

## 5 Conclusion

In the paper at hand, we introduced our study on the use of hybrid software and system development approaches. We presented the overall research design and then focused on the HELENA survey for which we presented the basic construction and initial findings obtained in HELENA's trial runs.

In the trials, we received 15 answers in total (including partial answers). Our primary goals of the trials were fulfilled, i.e., we could confirm that (1) the survey generates useful data and (2) the time required to fill in the questionnaire form is adequate. Even though the data collected in the test run cannot be used for a serious analysis, still, the data obtained shows some tendencies found in our previous research. In particular, we found mixed development approaches implemented in practice. Furthermore, we got insights into the implementation of traditional and agile practices in different project activities (e.g., project management or development/coding). We found for instance that agility is strongly represented in development-related activities, whereas management remains on a more traditional track. This also seems to confirm a hypothesis we made in [Th15], i.e., developers look for more agile/slim approaches to implement a system while managers prefer the classic way of working (see also [Mu13]). Reasons might be that managers, although being increasingly open for the agile mode of working, are constrained by stringent surrounding processes, external standards, or compliance requirements. Appropriate mechanisms to fully integrate agile processes into traditional processes and environments seem to be not mature enough yet. Also, changing the links between the agile processes and the surrounding processes might be needed, such as a closer cooperation with suppliers.

So far, we presented the research design and the first step concerning a large-scale study on the use of hybrid development approaches. Ongoing and future work comprises conducting the actual data collection. Currently, the first data collection is carried out, which will lead to further refinements of the survey instrument and corresponding more focused data collection steps and further surveys and interviews.

This paper is not only a status report of HELENA, it is also a call for participation: we cordially invite all practitioners to participate in our study and help us collecting and analyzing practically relevant data to improve the methods used for software and system development. Continuously updated information regarding the HELENA survey is available via ResearchGate: <https://goo.gl/1NzXmH>

## References

- [DPL16] Dikert, K.; Paasivaara, M.; Lassenius, C.: Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119:87-108, 2016.
- [Die15] Diebold, P.; Ostberg, J.-P.; Wagner, S.; Zandler, U.: What do practitioners vary in using scrum? In *International Conference XP*, pp. 40–51. Springer, 2015.
- [Din12] Dingsøy, T.; Nerur, S.; Balijepally, V.; Moe, N. B.: A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6):1213– 1221, 2012. Special Issue: Agile Development.

- [FK07] Fritzsche, M.; Keil, P.: Kategorisierung etablierter Vorgehensmodelle und ihre Verbreitung in der Deutschen Software-Industrie. Technische Universität München, Research Report (in German) TUM-I0717, 2007.
- [KF15] Kuhrmann, M.; Fernández, D. M.: Systematic software development: A state of the practice report from Germany. Proc. of Intl. Conf. on Global Software Engineering, IEEE, 2015.
- [KL15] Kuhrmann, M.; Linssen, O.: Vorgehensmodelle in Deutschland: Nutzung von 2006-2013 im Überblick. Gesellschaft für Informatik, MAW-Rundbrief, 39:32–47, 2015.
- [Mu13] Murphy, B.; Bird, C.; Zimmermann, T.; Williams, L.; Nagappan, N.; Begel, A.: Have agile techniques been the silver bullet for software development at Microsoft. Proc. of Intl. Symposium on Empirical Software Engineering and Measurement. ACM/IEEE, 2013.
- [Th15] Theocharis, G.; Kuhrmann, M.; Münch, J.; Diebold, P.: Is Water-Scrum-Fall Reality? On the Use of Agile and Traditional Development Practices. Proc. of Intl. Conf. on Product-Focused Software Process Improvement, Springer, ser. Lecture Notes in Computer Science 9459:149-166, 2015.
- [VerOne] VersionOne. State of agile survey. Available from: <http://www.versionone.com/agile-resources/more-resources/blogs/>, 2006-2015.
- [VB15] Vijayarathy, L.; Butler, C.: Choice of software development methodologies – do project, team and organizational characteristics matter? IEEE Software, (99):1ff., 2015.
- [We11] West, D.: Water-Scrum-Fall is the reality of agile for most organizations today. Technical report, Forrester, 2011.